High-Order Non-Oscillatory Schemes Using Meshfree Interpolating Moving Least Squares Reconstruction for Hyperbolic Conservation Laws

Warisa Yomsatieankul

2010

High-Order Non-Oscillatory Schemes Using Meshfree Interpolating Moving Least Squares Reconstruction for Hyperbolic Conservation Laws

Von der
Carl-Friedrich-Gauß-Fakultät
Technische Universität Carolo-Wilhelmina zu Braunschweig

zur Erlangung des Grades eines
Doktorin Naturwissenschaften (Dr.rer.nat.)
genehmigte

Dissertation

von

Warisa Yomsatieankul, MSc

geboren am 27.04.1975

aus Chonburi, Thailand

Eingereicht am:	15 Mai 2010
Mündliche Prüfung am:	31 Juli 2010
Referent	Prof. Dr. Thomas Sonar
Korreferent	Prof. Dr. Klaus-Jürgen Förster

2010

Abstract

Meshfree methods have attracted much attention for the development and their applications in the recent years. The methods are commonly formulated using the Moving Least Squares (MLS) methods. The interpolation version of the methods is determined by introducing the singular weight functions for constructing the shape functions and called as Interpolating Moving Least Squares (IMLS) methods. Since the shape functions of the IMLS interpolants satisfy the Kronecker delta, the IMLS methods have the property of nodal interpolation. For more information of the IMLS method the explicit formulae of the derivatives of the IMLS interpolants are derived. The methods are applied to a linear scalar conservation law with the Euler and Lax-Wendroff time discretizations.

The higher order schemes are presented employing a Taylor series expansion. The field variables and their successive derivatives are reconstructed using the IMLS methods. An analysis of the L_2-norm of this method is given. The Weighted Essentially Non-Oscillatory (WENO) schemes are adopted in the new schemes to prevent spurious oscillation. Our new methods based on staggered grids are discretized on space and the central Runge-Kutta schemes are used for time integration. Numerical results show that our new methods achieve the expected accuracy from an analysis of L_2-norm. Representative simulations show that the proposed methods are applicable to hyperbolic conservation laws.

Kurzfassung

In den letzten Jahren stieg das Interesse für die Entwicklung und Anwendung der netzfreien Verfahren an. Diese Verfahren basieren im Allgemeinen auf der Moving Least Square (MLS) Methode. Die interpolierende Form wird durch die Einführung einer singulären gewichteten Funktion bestimmt, um eine Kernfunktion, die so genannte IMLS Methode, aufzubauen. Die IMLS Methode mit ihrer Kernfunktion erfüllt die Kronecker Delta Eigenschaft und besitzt zugleich die Merkmale der Knoten-Interpolation. Um die IMLS Methode genauer zu untersuchen, wird die explizite Formel der IMLS Ableitung hergeleitet. Diese Verfahren finden in den linearen skalaren Erhaltungssätzen mit Euler- und Lax-Wendroff- Zeitintegration Anwendung.

Ein Verfahren höherer Ordnung wird über der Taylor Entwicklung vorgestellt. Dabei werden die Variablen und die aufeinander folgenden Ableitungen anhand der IMLS Methode rekonstruiert. Eine Analyse der L_2-Norm der Methode wird dargestellt. Die gewichteten wesentlich nichtoszillierenden Verfahren werden an die neue Methode angepasst um Oszillationen zu vermeiden. Das neue Verfahren, das auf versetzten Gittern basiert, wird für die Raum-Diskretisierung, und die zentrale Runge-Kutta Verfahren für die Zeit-Integration benutzt. Die numerischen Ergebnisse dieser Methode zeigen eine der L_2-Norm-Analyse entsprechende Genauigkeit. Repräsentative Simulationen zeigen, dass die vorgeschlagene Methode auf hyperbolische Erhaltungsgleichungen angewendet werden kann.

Acknowledgments

I would like to take the opportunity to thank all those who have helped, supported and inspired me during my doctoral study.

Foremost, I would like to express my deep and sincere gratitude to my supervisor Prof. Dr. Thomas Sonar for introducing me to work in the exciting field of research. His support, patience, and advice throughout the duration of this dissertation gave me motivation and guided me to develop understanding of hyperbolic conservation laws and meshfree methods. His guidance helped me in all time of research and writing of this dissertation. I would also like to thank Prof. Dr. Klaus-Jürgen Förster for having accepted to be my second referee.

My sincere thanks also goes to Thai government for the schlorship which enabled the financial support.

Lastly, and most importantly, I wish to thank my parents for their love and encouragement.

Contents

Tables,figures

Tables

Figures

Chapter 1

Introduction

1.1 Hyperbolic conservation laws

Hyperbolic conservation laws in one dimension are partial differential equations of the form

$$\frac{\partial u}{\partial t} + \frac{\partial}{\partial x} f(u) = 0 \tag{1.1}$$

where $u(x,t) \in \mathbb{R}^s$ is a s-dimensional vector of unknown functions and $f : \mathbb{R}^s \to \mathbb{R}^s$ is a vector of *flux function*. If all eigenvalues of Jacobian matrix $f'(u)$ are real and distinct, then the system (1.1) is called *strictly hyperbolic*. Integrating (1.1) with respect to x and t from x_1 to x_2 and t_1 to t_2 respectively, the result is

$$\int_{x_1}^{x_2} u(x,t_2)dx - \int_{x_1}^{x_2} u(x,t_1)dx = -\left(\int_{t_1}^{t_2} f(u(x_2,t))dt - \int_{t_1}^{t_2} f(u(x_1,t))dt\right). \tag{1.2}$$

The above equation is called the *integral form of conservation law* (1.1). The physical meaning of integral conservation (1.2) is that the change in the amount of field variable u in the interval $[x_1, x_2]$ between t_1 and t_2 is corresponding to the flux of the field variable across the boundaries $x = x_1$ and $x = x_2$ during the time interval from $t = t_1$ to $t = t_2$.
As some examples of conservation laws in this study, the following equations will be discussed.

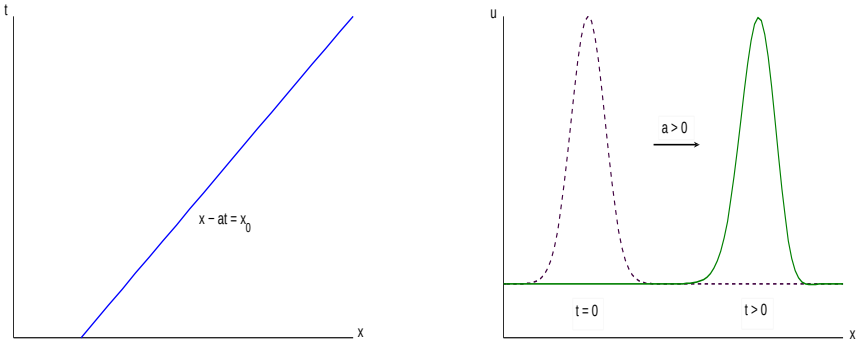
1. Linear advection equation
   A simple example of a scalar conservation law is a linear advection equation which is obtained by setting the flux $f(u) = au$ in (1.1). Rewrite the equation as

   $$u_t + au_x = 0 \tag{1.3}$$

   and assume that the initial condition is smooth $u(x,0) = u_0(x)$. Then the exact solution of (1.3) is simply obtained by method of characteristics [MM94], [LeV99]. The solution is

   $$u(x,t) = u_0(x - at). \tag{1.4}$$

   for $t \geq 0$. This initial data moves to the right when $a > 0$ and to the left when $a < 0$ with the velocity a. The solution $u(x,t)$ is constant along the characteristics which obey the equation $x - at = x_0$. These ideas are illustrated

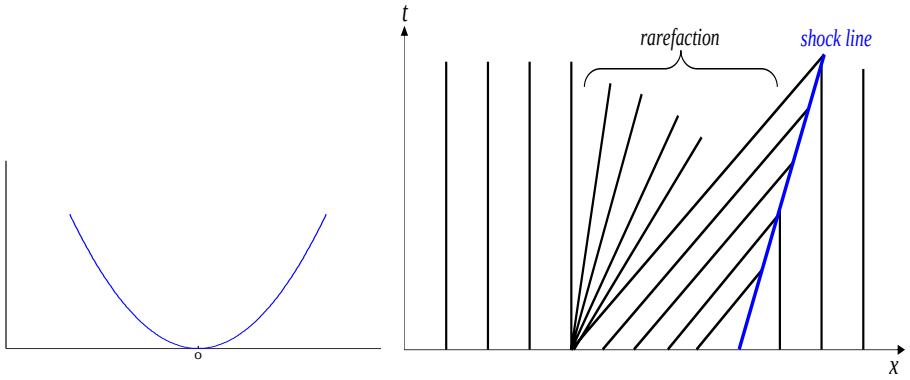


(a) Characteristic line (b) Evolution

FIGURE 1.1 *Characteristics in linear advection equation*

in Figure 1.1

2. Burgers' equation
   Burgers' equation is the most famous model problem in nonlinear scalar hyperbolic conservation laws. Assume $f(u)$ in (1.1) is a nonlinear function of u and f is a convex function, $f''(u) > 0$ for all u, the well-known nonlinear scalar Burgers' equation is demonstrated by setting $f(u) = \frac{1}{2}u^2$ in (1.1). Then (1.1) becomes
   $$\frac{\partial u}{\partial t} + \frac{\partial}{\partial x}\left(\frac{1}{2}u^2\right) = 0. \tag{1.5}$$
   This above equation is called *inviscid Burgers' equation*. In this case, the characteristic speed is equal to u. For limit time, a solution can be obtained by method of characteristics, i.e., $u(x,t) = u_0(x - tu(x,t))$ where u_0 is the initial data. But for increasing time the solution may be not unique because of the behaviour of the characteristics. When the characteristics first intersect at time T_b, the function $u(x,t)$ has no classical solution of hyperbolic PDE. The time at which the characteristics first intersect at $t = T_b$ is called the *breaking point* and given as $T_b = -\frac{1}{\min u_0'(x)}$. The weak solution becomes discontinuities. If $u_l > \frac{dx}{dt} > u_r$ where $\frac{dx}{dt}$ is the *speed of propagation of the discontinuity*, a discontinuity will be a *shock*. The *jump condition* or the *Rankine-Hugonoit (RH) condition* is a condition that the weak solution must satisfy across the RH condition. This condition is given as
   $$\frac{dx}{dt} = \frac{f(u_l) - f(u_r)}{u_l - u_r}. \tag{1.6}$$
   Another weak solution is the *rarefaction wave* in which the discontinuity propagates with the speed $\frac{dx}{dt}$. We display an example of the behaviour of Burgers'



(a) Convex flux (b) characteristics

FIGURE 1.2 *Burgers' equation*

equation in Figure 1.2.

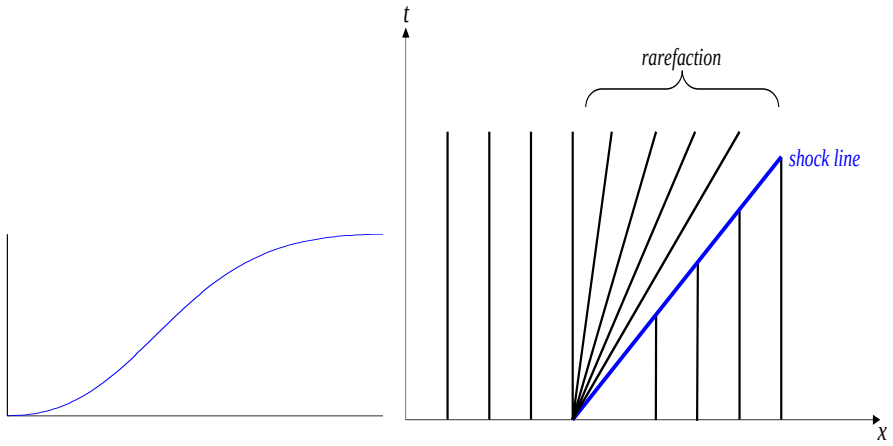
By introducing the *vanishing viscosity*, $\epsilon \frac{\partial^2 u}{\partial x^2}$, to the right hand side of (1.5), one can obtain the correct physical solutions. Another approach to find the solution of Burgers' equation in the limit $\epsilon \to 0$ is due to the concept of *entropy conditions*. The detailed description about the solutions of Burgers' equation can be founded in [LeV99, Tho99].

3. Buckley-Leverett equation.
   It is another form of nonlinear hyperbolic problems in which the flux function is non-convex as shown in Figure 1.3(a). The flux function we use in this study is of the form

$$f(u) = \frac{u^2}{u^2 + a(1-u)^2}, \tag{1.7}$$

   where a is a constant. Figure 1.3(b) shows the sketch of the characteristics for Buckley-Leverett equation with the initial values $u_l = 1$ and $u_r = 0$.

4. Euler equations.
   Euler equations govern physical behavior of the conservation laws of mass, momentum, and energy in the field of gas dynamics. The formulation of Euler equations in differential form can basically be classified by the choice of the dependent variables, i.e., the conservative variables, the characteristic variables, and the primitive variable variables.



(a) Non-convex flux (b) Characteristics [LeV99]

FIGURE 1.3 *Buckley-Leverett equation*

In conservation form, the equations are expressed in the vectors of conservative variables composed by density, momentum, and total energy which are written by using the vector notation as

$$\frac{\partial \mathbf{u}}{\partial t} + \frac{\partial}{\partial x}\mathbf{f}(\mathbf{u}) = 0, \tag{1.8}$$

where the conserved quantities $\mathbf{u}$ and $\mathbf{f}(\mathbf{u})$ are as follows $\mathbf{u} = \begin{pmatrix} \rho \\ \rho u \\ \rho e \end{pmatrix}$ and $\mathbf{f}(\mathbf{u}) = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ u(\rho e + p) \end{pmatrix}$, where ρ is density, u is velocity, p is pressure, and e is energy. Next, the equation of state is needed to calculate the pressure associated with the density, velocity, and enegy which can be specified as

$$p = (\gamma - 1)\left(\rho e - \frac{1}{2}\rho u^2\right) \tag{1.9}$$

with the ratio of specific heat $\gamma = 1.4$. In order to transform the conservation form to characteristic form, it is nescessary to rewrite (1.8) into quasi-linear form

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{A}\frac{\partial}{\partial x}\mathbf{u} = 0, \tag{1.10}$$

where $\mathbf{A} = \frac{\partial \mathbf{f}(\mathbf{u})}{\partial \mathbf{u}}$ is the *flux Jacobian matrix* and is given by

$$\mathbf{A} = \begin{pmatrix} 0 & 1 & 0 \\ \frac{\gamma-3}{2}u^2 & (3-\gamma)u & \gamma - 1 \\ -\gamma u e + (\gamma - 1)u^3 & \gamma e - \frac{3}{2}(\gamma - 1)u^2 & \gamma u \end{pmatrix}. \tag{1.11}$$

Using the relation $\mathbf{f}'(\mathbf{u}) = \mathbf{A}(\mathbf{u}) = QVQ^{-1}$ where

$$Q^{-1} = \begin{pmatrix} 1 - \frac{\gamma-1}{2}\frac{u^2}{c^2} & (\gamma-1)\frac{u}{c^2} & -\frac{\gamma-1}{c^2} \\ \frac{1}{\rho c}\left(\frac{\gamma-1}{2}u^2 - uc\right) & \frac{1}{\rho c}(c - u(\gamma-1)) & \frac{\gamma-1}{\rho c} \\ -\frac{1}{\rho c}\left(\frac{\gamma-1}{2}u^2 + uc\right) & \frac{1}{\rho c}(c + u(\gamma-1)) & -\frac{\gamma-1}{\rho c} \end{pmatrix}$$

$$Q = \begin{pmatrix} 1 & \frac{\rho}{2c} & -\frac{\rho}{2c} \\ u & \frac{\rho}{2c}(u+c) & -\frac{\rho}{2c}(u-c) \\ \frac{u^2}{2} & \frac{\rho}{2c}\left(\frac{u^2}{2} + uc + \frac{c^2}{\gamma-1}\right) & -\frac{\rho}{2c}\left(\frac{u^2}{2} - uc + \frac{c^2}{\gamma-1}\right) \end{pmatrix}$$

$V = \begin{vmatrix} u & 0 & 0 \\ 0 & u+c & 0 \\ 0 & 0 & u-c \end{vmatrix}$ and c is the speed of sound $c^2 = \frac{\gamma p}{\rho}$, then the new form of the Euler equations is formulated by setting $\mathbf{w} = Q^{-1}\mathbf{u}$

$$\frac{\partial \mathbf{w}}{\partial t} + V\frac{\partial}{\partial x}\mathbf{w} = 0, \tag{1.12}$$

and these equations have been separated into three scalar equations of the form

$$\frac{\partial w_i}{\partial t} + v_i\frac{\partial}{\partial x}w_i = 0. \tag{1.13}$$

The elements of $\mathbf{w}$ are known as *characteristic variables*. Each characteristic variable satisfies the linear advection equation with the velocity which is given by the eigenvalues, λ_i, of the flux Jacobian matrix.

The last form of the Euler equations is described in the *primitive variables* that are density, pressure, and velocity. By defining the vector of primitive variables as $R = \begin{pmatrix} \rho \\ u \\ p \end{pmatrix}$, then the primitive form of the Euler equations can be written as

$$\frac{\partial R}{\partial t} + C\frac{\partial R}{\partial x} = 0, \tag{1.14}$$

where

$$C = M^{-1}\mathbf{A}M = \begin{vmatrix} u & \rho & 0 \\ 0 & u & \frac{1}{\rho} \\ 0 & \rho c^2 & u \end{vmatrix}.$$

And the matrices M^{-1} and M are given by

$$M = \begin{pmatrix} 1 & 0 & 0 \\ u & \rho & 0 \\ \frac{u^2}{2} & \rho u & \frac{1}{\gamma-1} \end{pmatrix}$$

$$M^{-1} = \begin{pmatrix} 1 & 0 & 0 \\ -\frac{u}{\rho} & \frac{1}{\rho} & 0 \\ \frac{(\gamma-1)u^2}{2} & -(\gamma-1)u & \gamma-1 \end{pmatrix}.$$

More information and details about hyperbolic conservation laws can be founded in [GR96, Hir90, Lan98, LeV99, LeV02, Tho99].

1.2 Discretization

Numerical solutions of the differential equations are usually formulated by discretizing in both space and time.

1. Time integration.
   Rewrite (1.1) as
   $$u_t = L(v) = -f(u)_x, \tag{1.15}$$
   in case of time integration we substitute the partial differential operators for the time derivative in (1.15) with difference expression as follows:

   (a) First order Euler method
   $$u_i^{n+1} = u_i^n + \Delta t L_i(u^n). \tag{1.16}$$
   This method will be studied in Chapter 4, section 4.2.

   (b) Second order Lax-Wendroff method
   We derive the Lax-Wendroff method by expanding a Taylor series in the variable t, giving
   $$u(x, t+\Delta t) = u(x,t) + \Delta t u_t(x,t) + \frac{\Delta t^2}{2} u_{tt}(x,t) + O(\Delta t)^3. \tag{1.17}$$
   Then we convert the t-derivatives into x-derivatives by using the differential equation (1.1). This method will be studied in Chapter 4, section 4.2.

   (c) The second and the third order TVB Runge-Kutta scheme
   The details of both schemes will be presented in Chapter 6, section 6.3.

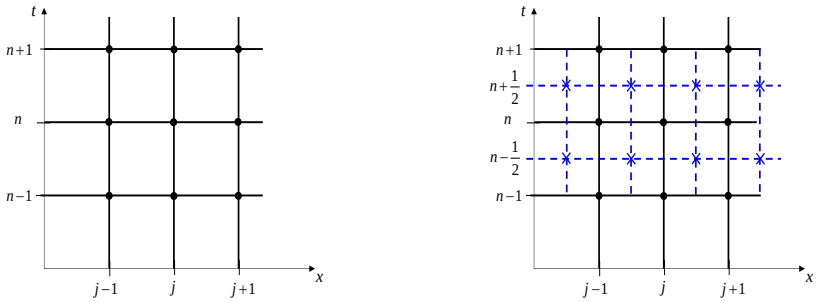
2. CFL condition.
   CFL condition is a necessary but not sufficient condition for numerical stability. This condition was first introduced by Courant, Friedrichs, and Lewy and was applied to FDMs for PDEs in 1928. The condition states that the domain of dependence of the discrete problem includes the domain of dependence of PDEs as the size of difference steps goes to zero [LeV99]. For the numerical

schemes based on non-stagered grid see Figure 1.4(a), the CFL condition

$$\lambda = \left| a\frac{\Delta t}{h} \right| \leq 1, \tag{1.18}$$

must be satisfies where $a = \max_{1 \leq i \leq d} |v_i|$ is the spectral radius of Jacobian matrix $\mathbf{A}$ in (1.11), v_i denoting the i-th eigenvalue. In case of the methods based on staggered grid see Figure 1.4(b), the CFL condition is

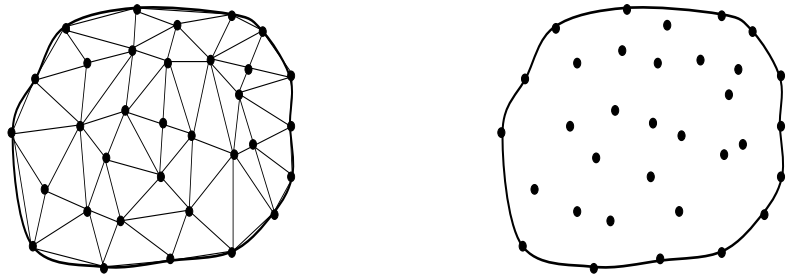
$$\lambda = \left| a\frac{\Delta t}{h} \right| \leq \frac{1}{2}. \tag{1.19}$$



(a) Non-staggered grid (b) Staggered grid

FIGURE 1.4 *Discretization in space and time*

3. Spatial discretization.
   The classical famous numerical schemes which depend on mesh-based discretization methods such as FDM (Finite Difference Method), FEM (Finite Element Method), and FVM (Finite Volume Method) have been widely used in science and engineering. In these methods, the domain of problem is discretized into meshes. Although these traditional mesh-based methods are robust and have been thoroughly developed for many engineering problems, there are still some limitations. For example, problems with distorted meshes require adaptive procedures which are very costly and time consuming. Therefore, the new method for decreasing or at least facilitating this limitation is introduced in the concept of *Meshfree or Meshless method*. More detail of this method will be presented in the next section.



(a) Mesh-based method (b) Meshfree method

FIGURE 1.5 *Spatial discretization*

1.3 Meshfree Method

Meshfree methods for numerical analysis and computation of partial differential equations are popularly known and were developed over the last decade. The methods approximate partial differential equations based only on a set of nodes without the need for an additional mesh, see Figure 1.5(b). Traditional mesh-based methods, such as finite difference, finite element, or finite volume methods, require a predefinition of the node connectivity because they depend on the meshes, see Figure 1.5(a). Creation and/or remeshing of these meshes becomes a difficult task for high-dimensional problems and for the complex geometry for the domain of interest. Mesh generators need a large consuming time of any mesh-based methods. Because meshfree methods are based only on a set of independent nodes, the costs of mesh generations are eliminated. This is why meshfree methods become an alternative choice of the numerical tools. Nowadays there is a huge number of different meshfree methods but here we only give a brief overview of the common methods. A good overview of meshfree methods is given in [BKFK96, Fri03].

The definition of meshfree method is given by Idelsohn et al. [IOCP03] as follows

Definition 1. *A meshless or meshfree method is an algorithm that satisfies both of the following statements:*

1. *the definition of the shape functions depends only on the node positions.*
2. *the evaluation of the nodes connectivity is bounded in time and its depends exclusively on the total number of nodes in the domain.*

In practice meshfree methods are more time consuming than mesh-based methods. Because for every integration node on the domain of interest the neighboring nodes have to be determined and several operations are necessary to compute the shape functions. The construction of the shape functions can mainly seperate into two groups. First, the shape functions are constructed by using Moving Least Squares (MLS) methods which are firstly introduced by Lancaster and Salkauskas [LS81]. The other construction is that it uses the Reproducing Kernel Particle Methods (RKPM) for the determination of the shape functions. The RKPM have been introduced by Liu *et al.* [LJZ95]. The concept of the MLS method is using a local least squares approximation around a fixed point. In general, the RKPM are a class of operators that reproduce the function itself through integration over the domain [Fri03]. An overview of the RKPM is given in [LCJ+96].

Based on the computational formulation, meshfree methods can be largely grouped for solving partial differential equations into three categories as follows:

1.3.1 Method based on strong form

An attractive property of this formulation is that it is simple to code and to implement because of missing requirement of integration. Moreover, it is really meshfree because it requires no mesh generators. The following methos are some of meshfree methods in the strong form.

1. Smooth Particle Hydrodynamics (SPH)
   The SPH was invented by Lucy [Luc93], Gingold and Monagham [GM77] to model astrophysical problems. The SPH is a meshfree particle-based method in which discrete particle properties are smoothed by a kernel function. The approximation of functions is done using a quadrature rule approximating integrals [Nid05]. A major attractive of the SPH techniques is that the requirement of computational grids is removed when computing spatial derivatives. Instead, an estimate of the derivatives is found by an exact derivative of an approximate function [Mon05]. A drawback of the classical SPH is that it is not able to reproduce constant function on the entire domain because the method has no the property of partition of unity. This problem was noticed by Liu *et al.* [LJZ95]. They improved the method to fulfill the consistency, or the completeness with the use of a correction function and called as a *corrective* SPH. The concept of a corrective SPH is to construct a corrective kernel, a product of the corrective function with the original kernel [LL02]. This new method is konwn as the RKPM. The SPH techniques have been applied to several classes of problems such as surface flow [Mon94, MFZ97], explosion phenomenon [LLL97], heat conduction [CM97, CM99, ZB04], etc.

2. Meshfree collocation method
   Several techniques have been proposed in the meshfree collocation methods. The Finite Point Method (FPM) [OIZT96] is based on a weigth least square interpolation of point data to construct the smooth approximations of functions. The Radial Point Collocation Method (RPCM) [WL02] is formulated using Gaussian and multiquadric radial basis functions [Kan90a]. The numerical techniques through radial basis functions have been developed in the application of partial differential equations by Kansa [Kan90a, Kan90b]. More information about radial basis functions and their applications in a class of meshfree methods are given in [Fas07]. Another approach of meshfree collocation methods based on the RKPM for constructing the shape functions is given in [Alu00, AL01].

3. Generalized Finite Difference Method (GFDM) or Meshfree Difference Method (MDM)
   The GFDM [LO80, LDT96] can be considered as another group of meshless method which establish the discrete equations directly from differential equations. In [LO80] the author improved the flexibility of conventional FDM in constructing difference schemes. Nevertheless the discrete equations yielded by the GFDM do not have the favourable properties for example well-conditioned, symmetric, positive definite. The improvement of the GFDM is given in [LC02]. The discrete equations are determined by minimizing a global residual which is the sum of residuals from all the nodes scattered in the problem domain [LC02]. The work in [BUG01] shows explicit formulae for the GFDM using irregular grids. The applications of the GFDM to the explicit solution of parabolic and hyperbolic equations are given in [BUG07]. The positive schemes of a generalization of the central difference has been introduced in [FS01]. This scheme is based on adding an explicit artificial viscosity to hyperbolic conservation laws.

1.3.2 Method based on weak form

The methods from this formulation are not truly meshfree methods because it requires mesh for the integration of the weak form. Some methods are given as follows:

1. Diffuse Element Method (DEM) or Diffuse Approximation Method (DAM)
   This approach is presented as a generalization of the FEM. The use of the MLS approximations for constructing the meshfree shape functions has been firstly developed by Nayroles *et al.* [NTV92] for numerical solutions of some partial differential equations in the weak form. It is called as the DEM or the DAM. The main advantage of the DEM is that it uses only a set of nodes and thus avoids finite element mesh generation. However the disadvantage of this approach is the incomplete derivatives of their basis functions, see [Gho10].

With the use of the complete derivatives in the methods, the accuracy of the numerical results is improved. Therefore, the *Diffuse Element Kansa Method* (DEKM) [Gho10] was introduced to construct the complete basis by using radial basis functions. As an application of the methods, one can find in [MCM77] for electromagnatic field problem, in [SP96] for fluid flow and heat transfer problems.

2. Element Free Galerkin Method (EFGM)
   Belytschko and his coworkers [BLG96] generalized the DEM and presented the EFGM. The EFGM uses the MLS approximations to compute the trial and test functions for the weak form. The difference between the DEM and the EFGM is the terms in the derivatives of the shape functions, see [Gho10]. In general, the problems with the MLS approximants is that they do not have the property of nodal interpolants, i.e., $\phi_i(\mathbf{x}_j) = \delta_{ij}$ where $\phi_i(\mathbf{x}_j)$ is the shape function corresponding to the node at $\mathbf{x}_i$, evaluated at a point $\mathbf{x}_j$ and δ_{ij} is the Kronecker delta. Therefore, the idea of essential boundary conditions is introduced in the EFGM. Several methods have been studied to treat the boundary problems such as Lagrange multipliers method [BLG96, KB95], the direct collocation methods [LBG92, BT96, ZA98], and the combination of FEM and EFGM [BOK95, KB96, Heg96].

3. Meshless Local Petrov-Galerkin (MLPG) method
   The MLPG is one of different approaches which proposed to eliminate the need of background mesh besides the collocation method. Atluri and Zhu [AZ98] proposed a general framework for developing the MLPG method. This approach is flexible to choose the local weak forms, the trial functions, and the test functions for solving partial differential equations. The advantages of the MLPG methods are as follows: (1) all weak forms are formulated locally; (2) various trial and test functions can be chosen and combined together for solving one problem; (3) overlapping local sub-domains can be chosen in such a way as to match problems and algorithms in any special cases [HLRA06]. The MLPG techniques have been employed to many problems such as the large deformation problems [ZYZ06], incompressible Navier-Stokes equations [LA01].

1.3.3 Method based on weak-strong form (MWS)

The last form takes the usefulness of the two above strategies in order to obtain the stable and accurate results and to use the minimum numbers of cells for numerical integration. For the spatial discretization, the strong form is performed for all nodes in the internal domain and the nodes on the essential boundaries; the weak form is used for the nodes near or on the boundaries. This idea has been proposed by Liu's group for linear solid mechanics [LG04] and for time dependent problems [GL05].

1.4 Motivation and Outline of this study

In recent years, the development of high resolution methods for the approximate solutions of hyperbolic conservation laws was achieved by many researchers. A review and an analysis of such methods are found, for example, in [GR96, Hir90, Lan98, LeV99].

The Essentially Non-Oscillatory (ENO) scheme was first proposed by Harten *et al.* [HEOC87]. This scheme is one of high order schemes which is based on a reconstruction technique. Its strategy is to select a best candidate stencil in each cell of the grid. The procedure of stencil selection is fulfilled by minimizing the oscillations due to the approximation of the given function and its derivatives. In ENO scheme to increase the accuracy of the scheme, it is necessary to increase the degree of the interpolating polynomial. However, there is a possible way to construct high order schemes with lower degree polynomials. This idea is at the heart of the Weighted Essentially Non-Oscillatory (WENO) scheme [LOC94, JS96] by taking a convex combination of all the candidate stencils instead of selecting the best one.

The advantage of high order reconstruction with meshfree methods over the classical high order schemes is that this technique is suitable to construct an efficient and high order accurate schemes not only on structured grids but also on unstructured grids, and especially to evaluate higher order derivatives [CFCF$^+$06]. The development of high order reconstruction schemes based on meshfree methods has been proposed by Felgueroso *et al.* [CFCF$^+$06, CFCN$^+$07]. They applied the MLS approximations to high order finite volume schemes on unstructured grids for solving shallow water dynamics [CFCF$^+$06] and the compressible Navier-Stokes equations [CFCN$^+$07]. In addition another meshfree approach, specifically Reproducing Kernel Methods (RKM) [LLB97, LHC$^+$97, LL99], has also been introduced by Felgueroso *et al.* for solving acoustic problems [CFC08].

Although these meshfree strategies - MLS and RKM - used in the combination with finite volume discretizations produce an accurate numerical scheme, see [CFCF$^+$06, CFCN$^+$07, CFC08], the difficulty and the drawback of these approaches are associated with the evaluation of higher derivatives of meshfree shape functions. See details in Chapter 2, section 2.5. Then in the work of Felgueroso *et al.* [CFCF$^+$06, CFCN$^+$07, CFC08, NCFC$^+$08] the authors avoided this drawback by the help of the diffuse derivative to compute the second derivative of MLS approximations. However, it turns out that the cost of the computation of the first derivative MLS approximant is more expensive than the IMLS interpolant see Chapter 2, section 2.4. Because for this reason, our approach seems to be an alternative proper choice for

high order reconstruction and will be presented in Chapter 5.

The plan of this study is the following.

In Chapter 2, we begin with a description of MLS approximations mostly found in the literature. Another approach of the approximations will also be presented. It is formulated as a linearly constrained quadratic minimization problem. This approach is well known as *the Backus-Gilbert method*. For an interpolating case, we follow the work of Kunle [Kun01]. He derived how to approximate the interpolants and their derivatives with the help of Singular Value Decomposition (SVD). In order to apply the IMLS interpolants to hyperbolic conservation laws which is the main part of this study, an extension of the IMLS method is presented by introducing non-symmetric window functions for constructing the interpolants.

To investigate the behaviour of the interpolants and their derivatives, we exactly derive the difference formulae as Sonar [Son05] did from the Sherpard method. We present the derivation of the formulae of the IMLS interpolants in Chapter 3.

As an application of the derivatives from the IMLS method we verify accuracy with a standard test - Franke's function - for interpolation problem. Later we apply these derivatives to linear advection equations with two different window functions. Von Neumann analysis and the expression of amplification factors of the methods will be studied in Chapter 4.

In Chapter 5, we construct high order reconstruction schemes with the use of the derivatives of the IMLS interpolants. The method formulated with a piecewise linear functions is of the second order accuracy while the higher order schemes are obtained using a piecewise quadratic polynomial reconstruction. An analysis of the L_2 error of these schemes will also be given. To avoid the spurious oscillations near discontinuities from high order schemes, we apply the concept of Weighted Essentially Non-Oscillatory (WENO) schemes as in [JS96].

We then study applications of the results from Chapter 5 to hyperbolic conservation laws in Chapter 6. In particular, we focus on the central schemes based on staggered grids. For space discretization we use the high order reconstructions from Chapter 5 and for time integration Runge-Kutta schemes are applied. One of our second order schemes are mainly based on a second order Nessyahu Tadmor scheme [NT90]. For the higher order method we follow Central WENO (CWENO) schemes developed by Levy *et al.* [LPR99b].

In Chapter 7, to confirm an efficiency of our methods we show some numerical results. Scalar and systems of conservation laws are the test problems of this study for our new schemes in one dimension. Chapter 8 presents our conclusion and the future work.

Chapter 2

Meshfree methods: Moving Least Squares Approximation

The MLS method was first proposed by Lancaster and Salkauskas [LS81] for approximation of scattered data. The notion of MLS is to start with a weighted least squares method for any fixed point and then move this point over the entire domain. A short review of MLS method will be presented in Section 2.1 and its interpolating variant will be discussed in Section 2.4.

2.1 Moving Least Squares

This review of moving least squares methods is based on a book by Fasshauer [Fas07] and can be described as follows.

2.1.1 Standard interpretation of MLS methods

Let us consider data $(\mathbf{x}_i, f(\mathbf{x}_i)), i = 1, \ldots, N$ where $\mathbf{x}_i \in \Omega \subset \mathbb{R}^s$ and $f(\mathbf{x}_i) \in \mathbb{R}$ with arbitrary $s \geq 1$, and let P_d^s be the space of s-variate polynomials of degree d with basis $\{p_1, p_2, \ldots, p_m\}$. This space of s-variate polynomials has dimension $m = \binom{s+d}{d}$. The discrete weighted l_2 inner product is defined by

$$\langle f, g \rangle_{w(\mathbf{x})} = \sum_{i=1}^{N} f(\mathbf{x}_i) g(\mathbf{x}_i) w_i(\mathbf{x}), \quad \mathbf{x} \in \mathbb{R}^s \text{ fixed} \tag{2.1}$$

with scalar weights $w_i(\mathbf{x}) = w(\mathbf{x} - \mathbf{x}_i)$ $i = 1, \ldots, N$. The induced norm is of the form

$$\|f\|_{2,w}^2 = \sum_{i=1}^{N} f(\mathbf{x}_i)^2 w(\mathbf{x}_i). \tag{2.2}$$

The best approximation with respect to norm (2.2) is determined by

$$Lf(\mathbf{x}) = \sum_{j=1}^{m} c_j(\mathbf{x}) p_j(\mathbf{x}) \tag{2.3}$$

such that

$$\sum_{i=1}^{N} (Lf(\mathbf{x}_i) - f(\mathbf{x}_i))^2 w_i(\mathbf{x})$$

is minimized. One has to solve the normal equations

$$\sum_{j=1}^{m} c_j(\mathbf{x}) \langle p_j, p_k \rangle_{w(\mathbf{x})} = \langle f, p_k \rangle_{w(\mathbf{x})}, \qquad k = 1, \ldots, m, \tag{2.4}$$

or, in the matrix-vector notation,

$$G(\mathbf{x})\mathbf{c}(\mathbf{x}) = \mathbf{f}_p(\mathbf{x}), \tag{2.5}$$

where the matrix $G(\mathbf{x}) = \mathbf{P}^T \mathbf{w}(\mathbf{x}) \mathbf{P}$ with

$$\mathbf{P} = \begin{pmatrix} p_1(x_1) & p_2(x_1) & \ldots & p_m(x_1) \\ p_1(x_2) & p_2(x_2) & \ldots & p_m(x_2) \\ \vdots & \vdots & \ddots & \vdots \\ p_1(x_N) & p_2(x_N) & \ldots & p_m(x_N) \end{pmatrix},$$

and the weight matrix as

$$\begin{aligned} \mathbf{w}(\mathbf{x}) &= \mathsf{diag}(w(\mathbf{x} - \mathbf{x}_1), w(\mathbf{x} - \mathbf{x}_2), \ldots, w(\mathbf{x} - \mathbf{x}_N)) \\ &= \mathsf{diag}(w_1(\mathbf{x}), w_2(\mathbf{x}), \ldots, w_N(\mathbf{x})). \end{aligned}$$

The right hand side of (2.5) is defined as

$$\mathbf{f}_p(\mathbf{x}) = \left[\langle f, p_1 \rangle_{w(\mathbf{x})}, \cdots, \langle f, p_m \rangle_{w(\mathbf{x})} \right]^T. \tag{2.6}$$

If the number of nodal supports is less than the order of the basis function, then the Gram matrix $G(\mathbf{x})$ is singular and can not be inverted. Therefore the condition which makes Gram matrix $G(\mathbf{x})$ invertible is that the number of nodal supports must be greater than or equal to the dimension of the basis function, i.e., $N \geq m$.

The unknown coefficients $\mathbf{c}(\mathbf{x})$ in (2.5) are then obtained as

$$\mathbf{c}(\mathbf{x}) = G^{-1}(\mathbf{x})\mathbf{f}_p(\mathbf{x}). \tag{2.7}$$

Substituting (2.7) into (2.3), the approximation can be rewritten in the following form

$$Lf(\mathbf{x}) = \sum_{j=1}^{N} \phi_j(\mathbf{x}) f_j = \mathbf{\Phi}(\mathbf{x})\mathbf{f}, \tag{2.8}$$

where the vector of *shape* or *kernel* function $\mathbf{\Phi}(\mathbf{x})$ is defined in the form

$$\mathbf{\Phi}(\mathbf{x}) = \mathbf{w}(\mathbf{x})\mathbf{P}^T(\mathbf{x})G^{-1}(\mathbf{x})\mathbf{p}(\mathbf{x}). \tag{2.9}$$

2.1.2 The Backus-Gilbert approach

In the Backus-Gilbert approach, the approximant will be formulated as the quasi-interpolant

$$Lf(\mathbf{x}) = \sum_{i=1}^{N} f(\mathbf{x}_i)\psi_i(\mathbf{x}), \tag{2.10}$$

where $\mathbf{f} = [f(\mathbf{x}_1), \ldots, f(\mathbf{x}_N)]^T$ is given. In this approach, one attempts to find the values of generating functions, $\psi_i(\mathbf{x}) = \psi(\mathbf{x}, \mathbf{x}_i)$, by minimizing

$$\frac{1}{2}\sum_{i=1}^{N} \psi_i^2(\mathbf{x}) \frac{1}{w(\mathbf{x}, \mathbf{x}_i)} \tag{2.11}$$

subject to the constraints

$$\sum_{i=1}^{N} p(\mathbf{x}_i)\psi_i(\mathbf{x}) = p(\mathbf{x}) \qquad \forall p \in P_d^s. \tag{2.12}$$

The problem defined in (2.11) with polynomial reproduction constraints (2.12) can be written in matrix form as

$$\frac{1}{2}\Psi(\mathbf{x})^T Q(\mathbf{x})\Psi(\mathbf{x}) \tag{2.13}$$

and the constraints

$$A\Psi(\mathbf{x}) = \mathbf{p}(\mathbf{x}), \tag{2.14}$$

where $Q(\mathbf{x}) = \mathsf{diag}\left(\frac{1}{w(\mathbf{x},\mathbf{x}_1)}, \ldots, \frac{1}{w(\mathbf{x},\mathbf{x}_N)}\right)$, A is the $m \times N$ matrix with entries $A_{ji} = p_j(\mathbf{x}_i)$, $i = 1, \ldots, N$, $j = 1, \ldots, m$ and $\mathbf{p} = [p_1, \ldots, p_m]^T$ is a vector of polynomial basis for the space P_d^s of degree d. This problem is called a quadratic minimization problem and can be solved with the help of Lagrange multipliers, $\Lambda = (\lambda_1, \ldots, \lambda_m)^T$,

i.e., we consider solving the minimization of

$$\frac{1}{2}\Psi(\mathbf{x})^T Q(\mathbf{x})\Psi(\mathbf{x}) - \Lambda^T \left[A\Psi(\mathbf{x}) - \mathbf{p}(\mathbf{x})\right] \tag{2.15}$$

with respect to Ψ and Λ. Assume $Q(\mathbf{x})$ is invertible. The minimization of (2.15) is achieved by differentiating (2.15) with respect to Ψ and Λ and this leads to

$$Q\Psi - A^T\Lambda = \mathbf{0} \tag{2.16}$$

$$A\Psi - \mathbf{p} = \mathbf{0}, \tag{2.17}$$

or in matrix form

$$\begin{pmatrix} Q & -A^T \\ A & \mathbf{0} \end{pmatrix} \begin{pmatrix} \Psi \\ \Lambda \end{pmatrix} = \begin{pmatrix} \mathbf{0} \\ \mathbf{p} \end{pmatrix}. \tag{2.18}$$

Solving formally yields

$$\Lambda(\mathbf{x}) = \left(AQ^{-1}(\mathbf{x})A^T\right)^{-1} \mathbf{p}(\mathbf{x}) = G^{-1}(\mathbf{x})\mathbf{p}(\mathbf{x}) \tag{2.19}$$

$$\Psi(\mathbf{x}) = Q^{-1}(\mathbf{x})A^T\Lambda(\mathbf{x}), \tag{2.20}$$

or, in the componentwise form, (2.20) leads to

$$\Psi_i(\mathbf{x}) = w(\mathbf{x}, \mathbf{x}_i) \sum_{j=1}^{m} \lambda_j(\mathbf{x}) p_j(\mathbf{x}), \quad i = 1, \ldots, N. \tag{2.21}$$

2.1.3 Equivalence of standard MLS formulation and the Backus-Gilbert Approach

Now we present that the two formations of MLS approximation are equivalent that is we show that $Lf(\mathbf{x})$ in (2.3) and (2.10) are the same. Rewrite the approximant (2.3) in the vector notation

$$Lf(\mathbf{x}) = \sum_{j=1}^{m} c_j(\mathbf{x}) p_j(\mathbf{x}) = \mathbf{p}^T(\mathbf{x})\mathbf{c}(\mathbf{x}),$$

where $\mathbf{p} = (p_1, \ldots, p_m)^T$ and $\mathbf{c} = (c_1, \ldots, c_m)^T$. Recall that the unknown coefficients $\mathbf{c}(\mathbf{x})$ are given as

$$\mathbf{c}(\mathbf{x}) = G^{-1}(\mathbf{x})\mathbf{f}_p(\mathbf{x}).$$

The vector $\mathbf{f}_p(\mathbf{x})$ can be written as

$$\mathbf{f}_p(\mathbf{x}) = A(\mathbf{x})Q^{-1}(\mathbf{x})\mathbf{f}$$

where $Q^{-1}(\mathbf{x})$ is used in the Backus-Gilbert formuation. Then we have

$$\mathbf{c}(\mathbf{x}) = G^{-1}(\mathbf{x})A(\mathbf{x})Q^{-1}(\mathbf{x})\mathbf{f}. \tag{2.22}$$

Substituting (2.22) into (2.3), we get

$$Lf(\mathbf{x}) = \mathbf{p}^T(\mathbf{x})\mathbf{c}(\mathbf{x}) = \mathbf{p}^T(\mathbf{x})G^{-1}(\mathbf{x})A(\mathbf{x})Q^{-1}(\mathbf{x})\mathbf{f}. \tag{2.23}$$

For the Backus-Gilbert approach, the approximant $Lf(\mathbf{x})$ can be written in the vector notation as

$$Lf(\mathbf{x}) = \Psi^T(\mathbf{x})\mathbf{f} = (Q^{-1}(\mathbf{x})A^T(\mathbf{x})G^{-1}(\mathbf{x})\mathbf{p}(\mathbf{x}))^T\mathbf{f}. \tag{2.24}$$

By the symmetry of $Q(\mathbf{x})$ and $G(\mathbf{x})$, the above equation is the same as (2.23).

2.2 Window function

A *window function* or *weight function* plays an important role to distinguish the Moving Least Squares(MLS) from the Standard Least Squares approximation. The smoothness of window functions reflects directly on the MLS shape functions.

The type of window functions which is widely used in MLS methods is in the exponential functions and spline functions. A crucial parameter which is defined in the sense that the window function should be non-zero only over a small domain of influence is called *dilation parameter* or *the size of support domain*. This parameter defines the sparse linear systems of the discrete equations. The window functions should satisfy the following properties:

- Positivity: $w(\mathbf{x} - \mathbf{x}_i) \geq 0$.
- $w(\mathbf{x}-\mathbf{x}_i)$ is a monotonically increasing function as $|\mathbf{x} - \mathbf{x}_i|$ is decreasing. That is, the window functions have relatively large values for the nodal points close to the collocation point and relatively small for those away at some distance.
- Compactness of support: $w(\mathbf{x}-\mathbf{x}_i) = 0 \Leftrightarrow |\mathbf{x} - \mathbf{x}_i| > R$ where R is *the dilation parameter*.
- Partition of unity: $\sum_i w(\mathbf{x} - \mathbf{x}_i) = 1$.
- $\lim\limits_{\mathbf{x}\to\mathbf{x}_i} w(\mathbf{x} - \mathbf{x}_i) \longrightarrow \infty$, for the interpolating case.

In this work, the Gaussian function as in [AD01] will be used. For the purpose of improvement the MLS schemes need to diminish unphysical oscillations

in convection-dominated problems by means of upwind strategy. Hence a non-symmetric window function will also be discussed. Let us define the window function in one dimension

$$w(z) = \begin{cases} \frac{e^{-\delta\left(\frac{z}{R^*}\right)^2} - e^{-\delta}}{1-e^{\delta}} \left(1 - \left(\frac{z}{R^*}\right)^2\right)^4 & \text{if } -R^* < z < 0 \\ \frac{e^{-\delta\left(\frac{z}{R}\right)^2} - e^{-\delta}}{1-e^{\delta}} \left(1 - \left(\frac{z}{R}\right)^2\right)^4 & \text{if } 0 < R < 0 \\ 0 & \text{otherwise} \end{cases} \tag{2.25}$$

where $z = x - x_i$,

$$R^* = R \quad \text{for symmetric window function} \tag{2.26}$$

and

$$R^* = \min_{1 \le i \le N} |x - x_i| \quad \text{for a non-symmetric window function} \tag{2.27}$$

as shown in Figure 2.1. Based on the work by Lancaster [LS86], to compute an interpolating MLS version, weight functions with a singularity have to be used and an inverse of even power of the distance gives the proper order of the singularity. Then for the interpolating case, the weight function $\tilde{w}$ is defined by

$$\tilde{w}(r) = \frac{w(r)}{\|x - x_i\|^4}. \tag{2.28}$$

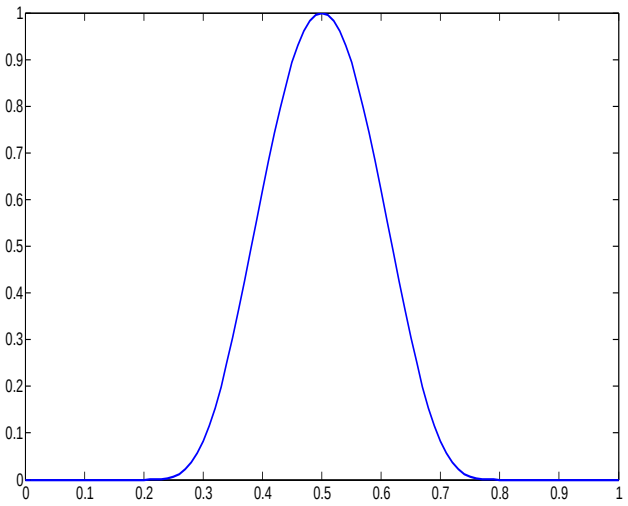
2.2.1 Dilation parameter

The size of dilation parameter R is designed in order to ensure that the normal matrix or Gram matrix $G(\mathbf{x})$ in (2.5) is invertible. In order to analyse the difference operators from the IMLS method it is a good choice to set the dilation parameter to be propertional to the product of the dimension of polynomial basis and the equi-distance h, i.e.,

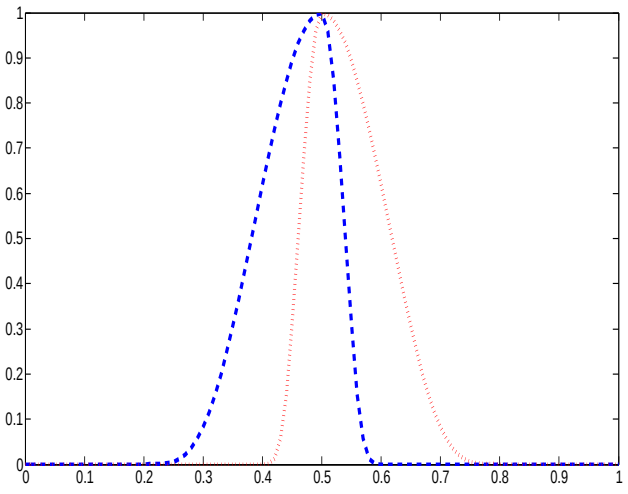
$$R = mh \quad \text{in the case of symmetric window function and} \tag{2.29}$$

$$R = (m+1)h \quad \text{in the case of non-symmetric window function.} \tag{2.30}$$

In this study, (2.29) and (2.30) are the minimum size of the dilation parameter to ensure invertibility of Gram matrix $G(\mathbf{x})$.



(a) Symmetric window function



(b) Non-symmetric window functions

FIGURE 2.1 *Different window functions*

2.3 Treatment of singularity

To achieve the interpolation version of MLS function, the weight function should be defined in the sense that the value of weights tends to ∞ around the singular node. According to (2.9), if the singular weight $\tilde{\mathbf{w}}$ is introduced into the MLS-shape function, then the Gram matrix in (2.9) can not be invertible. Kunle [Kun01] cured this singularity problem by adding a small value, ϵ, to the singular weight. In his work, this was done in a componentwise form. Later Netuzhylov et al. [NSY07], [Net08] have revised this singular treatment in the matrix-vector notation and extended it to derivatives of arbitrary order.
By adding the small value ϵ to (2.9), then (2.9) becomes

$$\mathbf{\Phi}(\mathbf{x}+\epsilon) = \tilde{\mathbf{w}}(\mathbf{x}+\epsilon)\mathbf{P}^T(x)G^{-1}(\mathbf{x}+\epsilon)\mathbf{p}(x). \tag{2.31}$$

where $G(\mathbf{x}+\epsilon) = \mathbf{P}^T\mathbf{w}(\mathbf{x}+\epsilon)\mathbf{p}$. In order to find the inverse matrix $G^{-1}(\mathbf{x}+\epsilon)$, we use the Sherman-Morrison Formula [PFTV92] which relates the matrix $G^{-1}(\mathbf{x}+\epsilon)$ to a given matrix $\tilde{G}^{-1}(\mathbf{x})$. Then the matrix $G^{-1}(\mathbf{x}+\epsilon)$ leads to

$$G^{-1}(\mathbf{x}_i+\epsilon) = \tilde{G}(\mathbf{x}_i)^{-1} - \frac{1}{\mathbf{p}^T(\mathbf{x}_i)\tilde{G}(\mathbf{x}_i)^{-1}\mathbf{p}(\mathbf{x}_i)} \left(\tilde{G}(\mathbf{x}_i)^{-1}\mathbf{p}(\mathbf{x}_i)\right)\left(\tilde{G}^{-1}\mathbf{p}(\mathbf{x}_i)\right)^T, \tag{2.32}$$

where

$$\tilde{G}(\mathbf{x}) = \begin{pmatrix} \sum_{\substack{j=1\\ j\neq i}}^{N} p_1(x_j)w_j(x_i)p_1(x_i) & \cdots & \sum_{\substack{j=1\\ j\neq i}}^{N} p_1(x_j)w_j(x_i)p_m(x_i) \\ \vdots & \ddots & \vdots \\ \sum_{\substack{j=1\\ j\neq i}}^{N} p_m(x_j)w_j(x_i)p_1(x_i) & \cdots & \sum_{\substack{j=1\\ j\neq i}}^{N} p_m(x_j)w_j(x_i)p_m(x_i) \end{pmatrix} \tag{2.33}$$

To find the inverse of $\tilde{G}(x)$, Kunle [Kun01] recommended Singular Value Decomposition(SVD) techniques.

2.4 IMLS shape functions and their derivatives

The interpolating version of MLS is achieved by introducing the singular weight function in to the MLS shape functions (2.9). The singularity can be avoided by the help of the regularization as mentioned in the previous section. In order to explicitly compute the expression of shape functions (2.4)

$$\mathbf{\Phi}(\mathbf{x}+\epsilon) = \tilde{\mathbf{w}}(\mathbf{x}+\epsilon)\mathbf{P}^T(x)G^{-1}(\mathbf{x}+\epsilon)\mathbf{p}(x),$$

we apply Taylor expansion around a point $\mathbf{x}$. For the details see [NSY07], [Net08]. The IMLS shape functions and their derivatives are summarized as follows:

Theorem 2.1. *For $\mathbf{x}_i = \mathbf{x}_j$ and $\alpha \geq 1$, the shape function and its derivaitves are given as*

$$\phi_i(\mathbf{x}_i) = 1, \tag{2.34}$$

$$\frac{d^\alpha}{d\mathbf{x}^\alpha}\phi_i(\mathbf{x}_i) = \frac{1}{\mathbf{p}^T(\mathbf{x}_i)\tilde{G}(\mathbf{x})^{-1}\mathbf{p}(\mathbf{x}_i)}\mathbf{p}^T(\mathbf{x}_i)\tilde{G}^{-1}\frac{d^\alpha}{d\mathbf{x}^\alpha}\mathbf{p}(\mathbf{x}_i) \tag{2.35}$$

where $\tilde{G}(\mathbf{x})$ is defined in (2.33).

Proof. See [NSY07], [Net08] for the proof. ■

Theorem 2.2. *For $\mathbf{x}_i \neq \mathbf{x}_j$ and $\alpha \geq 1$, the following equations*

$$\phi_j(\mathbf{x}_i) = 0, \tag{2.36}$$

$$\frac{d^\alpha}{d\mathbf{x}^\alpha}\phi_j(\mathbf{x}_i) = w_j(\mathbf{x}_i)\mathbf{p}^T(\mathbf{x}_j)\left[\tilde{G}^{-1}(\mathbf{x}_i) - \frac{\left(\tilde{G}^{-1}\mathbf{p}(\mathbf{x}_i)\right)\left(\tilde{G}^{-1}\mathbf{p}(\mathbf{x}_i)\right)^T}{\mathbf{p}^T(\mathbf{x}_i)\tilde{G}^{-1}\mathbf{p}(\mathbf{x}_i)}\right]\frac{d^\alpha}{d\mathbf{x}^\alpha}\mathbf{p}(\mathbf{x}_i) \tag{2.37}$$

hold.

Proof. See [NSY07], [Net08] for the proof. ■

2.5 MLS shape functions and their derivatives

Rewriting the transpose of the shape function (2.9) as

$$\mathbf{\Phi}^T(\mathbf{x}) = \mathbf{p}^T(\mathbf{x})G^{-1}(\mathbf{x})\mathbf{B}(\mathbf{x}), \tag{2.38}$$

where $\mathbf{B}(\mathbf{x}) = \mathbf{w}(\mathbf{x})\mathbf{P}(x)$, the first derivative of MLS shape functions can be directly computed by the product rule to give

$$\mathbf{\Phi}^T_{,k}(\mathbf{x}) = \mathbf{p}^T_{,k}(\mathbf{x})G^{-1}(\mathbf{x})\mathbf{B}(\mathbf{x}) + \mathbf{p}^T(\mathbf{x})G^{-1}_{,k}(\mathbf{x})\mathbf{B}(\mathbf{x}) + \mathbf{p}^T(\mathbf{x})G^{-1}(\mathbf{x})\mathbf{B}_{,k}(\mathbf{x}) \tag{2.39}$$

and the second derivatives then follow to

$$\begin{aligned}\mathbf{\Phi}^T_{,kl}(\mathbf{x}) = {} & \mathbf{p}^T_{,kl}(\mathbf{x})G^{-1}(\mathbf{x})\mathbf{B}(\mathbf{x}) + \mathbf{p}^T_{,k}(\mathbf{x})G^{-1}_{,l}(\mathbf{x})\mathbf{B}(\mathbf{x}) + \mathbf{p}^T_{,k}(\mathbf{x})G^{-1}(\mathbf{x})\mathbf{B}_{,l}(\mathbf{x}) \\ & + \mathbf{p}^T_{,l}(\mathbf{x})G^{-1}_{,k}(\mathbf{x})\mathbf{B}(\mathbf{x}) + \mathbf{p}^T(\mathbf{x})G^{-1}_{,kl}(\mathbf{x})\mathbf{B}(\mathbf{x}) + \mathbf{p}^T(\mathbf{x})G^{-1}_{,k}(\mathbf{x})\mathbf{B}_{,l}(\mathbf{x}) \\ & + \mathbf{p}^T_{,l}(\mathbf{x})G^{-1}(\mathbf{x})\mathbf{B}_{,k}(\mathbf{x}) + \mathbf{p}^T(\mathbf{x})G^{-1}_{,l}(\mathbf{x})\mathbf{B}_{,k}(\mathbf{x}) + \mathbf{p}^T(\mathbf{x})G^{-1}(\mathbf{x})\mathbf{B}_{,kl}(\mathbf{x})\end{aligned} \tag{2.40}$$

with

$$G^{-1}_{,k}(\mathbf{x}) = -G^{-1}(\mathbf{x})G_{,k}(\mathbf{x})G^{-1}(\mathbf{x})$$

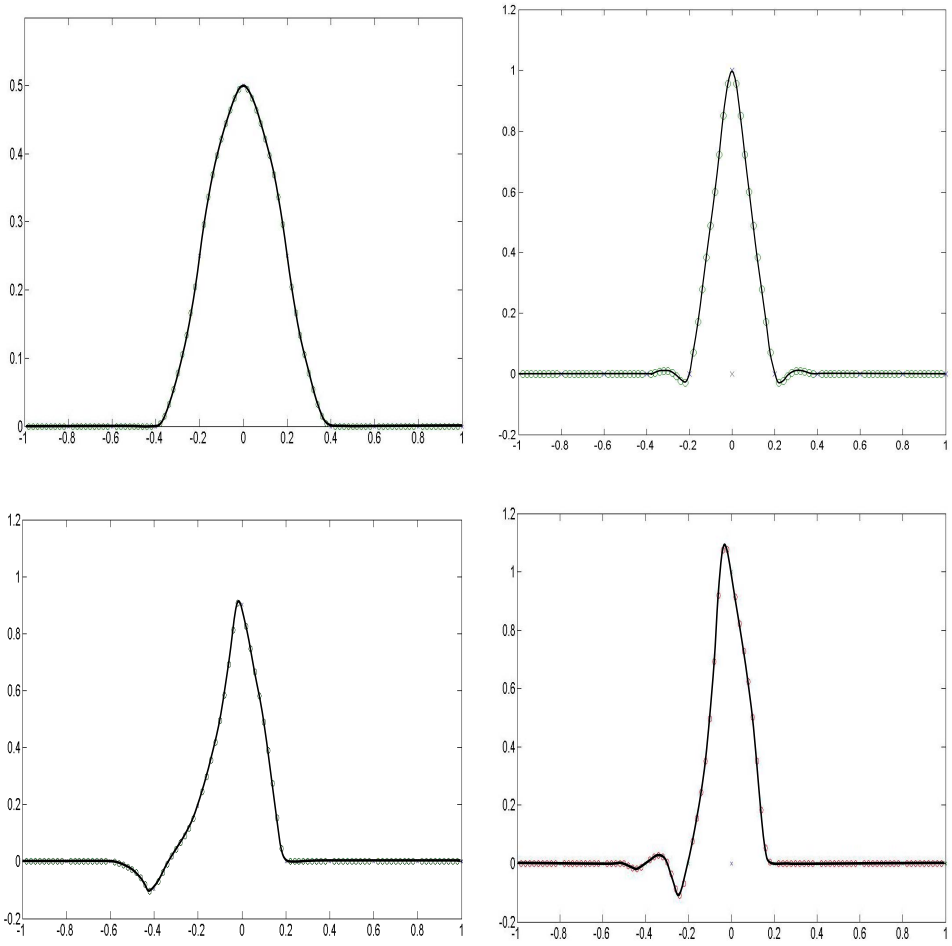
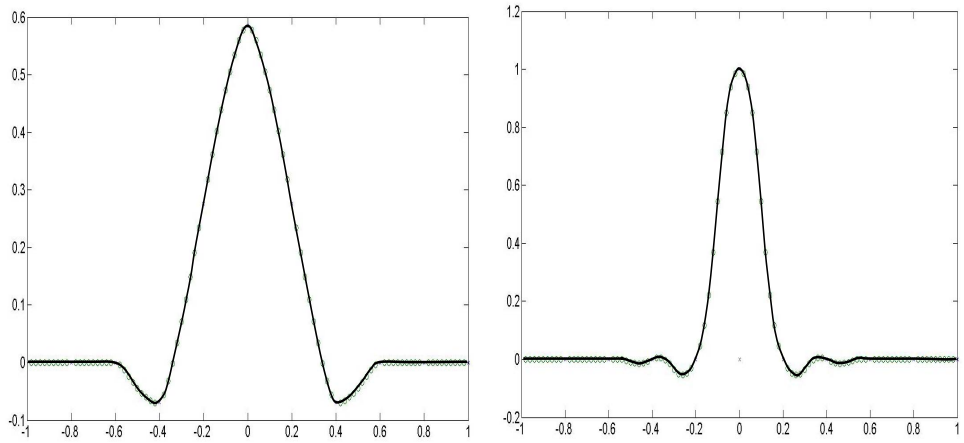


FIGURE 2.2 *Shape functions from MLS and IMLS method based on linear basis.*

and

$$G^{-1}_{,kl} = G^{-1}G_{,l}G^{-1}G_{,k}G^{-1} - G^{-1}G_{,kl}G^{-1} + G^{-1}G_{,k}G^{-1}G_{,l}G^{-1}.$$

As an example of the difference between MLS and IMLS methods, Figure 2.2 and Figure 2.3 show the shape functions located at specific point $x_j = 0$. For the linear method, the top of Figure 2.2 represent the result from employing symmetric window function while the bottom show the output from non-symmetric window function. For the quadratic method based on symmetric window function, the shape function is presented in Figure 2.3.



(a) MLS (b) IMLS

FIGURE 2.3 *Shape functions from MLS and IMLS methods based on quadratic basis.*

2.6 Properties of IMLS and MLS approximations

- Reproducing property
  According to (2.12),

$$\sum_{i=1}^{N} p(\mathbf{x}_i)\phi_i(\mathbf{x}) = p(\mathbf{x}) \qquad \forall p \in P_d^s.$$

  Hence, the MLS approximants reproduce the polynomial basis functions of degree d.

- Partition of unity
  The sum of shape functions is equal to one in the whole domain. Thus the shape functions build a partition of unity, i.e. $\sum_j \phi_j(x) = 1$.

- Partition of nullity
  The derivatives of the shape functions build a partition of nullities, i.e.,

$$\sum_j \nabla\phi_j(\mathbf{x}) = 0 = \sum_j \nabla^2\phi_j(\mathbf{x}).$$

- Interpolation properties
  In case of IMLS, the shape functions act as the Kronecker delta, that is

$$\phi_j(\mathbf{x}_i) = \begin{cases} 1 & \text{if } i = j, \\ 0 & \text{if } i \neq j. \end{cases}$$

  If the function $f(\mathbf{x})$ is approximated by IMLS methods at any fixed point

$\mathbf{x} = \mathbf{x}_i$, then

$$f(\mathbf{x}) = \sum_{j=1}^{N} \phi_j(\mathbf{x}_i) f_j = f(\mathbf{x}_i). \tag{2.41}$$

In all of this work, we only focus on the derivatives of IMLS approximations in one dimension. Let's define the notation of the IMLS shape functions corresponding to the different window functions in Section 2.2 as

1. $\frac{d}{dx}\phi^l(x)$ is the first derivative of IMLS interpolant based on the non-symmetric left window function.

2. $\frac{d}{dx}\phi^f(x)$ is the first derivative of IMLS interpolant based on the symmetric window function.

3. $\frac{d}{dx}\phi^r(x)$ is the first derivative of IMLS interpolant based on the non-symmetric right window function.

The second derivatives are analogously specified as above.

Chapter 3

Difference Formulae from IMLS method

In this chapter we analyse IMLS methods to derive the first and the second derivatives into explicit formulae as in the work of Sonar [Son05]. For simplicity we focus on one dimensional situations with the specific size of stencils depended on the magnitude of the dilation parameter R. Both, symmetric and non-symmetric window functions will be concerned.

3.1 Difference formulae of the first derivatives of interpolants

According to the expression of the derivatives of IMLS shape functions in Chapter 2, Section 2.4, the first derivative can be rewritten as

$$\frac{d}{dx}\phi_i(x_i) = \frac{1}{p^T(x_i)\tilde{G}(x_i)^{-1}p(x_i)} p^T(x_i)\tilde{G}(x_i)^{-1}\frac{d}{dx}p(x_i) \tag{3.1}$$

$$\frac{d}{dx}\phi_j(x_i) = w_j(x_i)p^T(x_j)\left[\tilde{G}(x_i)^{-1} - \frac{\left(\tilde{G}(x_i)^{-1}p(x_i)\right)\left(\tilde{G}(x_i)^{-1}p(x_i)\right)^T}{p^T(x_i)\tilde{G}(x_i)^{-1}p(x_i)}\right]\frac{d}{dx}p(x_i). \tag{3.2}$$

If the function $f(x)$ is interpolated by the IMLS method

$$Lf(x) = \sum_{j=1}^{N}\phi_j(x)f_j,$$

then the first derivative of the function $Lf(x)$ is

$$\frac{d}{dx}Lf(x) = \sum_{j=1}^{N}\frac{d}{dx}\phi_j(x)f_j \tag{3.3}$$

3.1.1 Linear basis

Substituting $p(x) = \{1, x\}$ and $\frac{d}{dx}p(x) = \{0, 1\}$ into (3.1) and (3.2), the first derivative of the IMLS shape functions can be rewritten as

$$\frac{d}{dx}\phi_i(x_i) = \frac{\sum\limits_{j,j\neq i} w_j(x_i - x_j)}{x_i^2 \sum\limits_{j,j\neq i} w_j - 2x_i \sum\limits_{j,j\neq i} w_j x_j + \sum\limits_{j,j\neq i} w_j x_j^2} \tag{3.4}$$

for $x_i = x_j$, and

$$\frac{d}{dx}\phi_j(x_i) = w_j \frac{x_j - x_i}{x_i^2 \sum\limits_{j,j\neq i} w_j - 2x_i \sum\limits_{j,j\neq i} w_j x_j + \sum\limits_{j,j\neq i} w_j x_j^2}. \tag{3.5}$$

for $x_i \neq x_j$. To avoid the difficulty, we only consider a uniform grid, i.e., $x_{i+1} - x_i = h$.

Lemma 3.1. *The explicit difference formulae from IMLS method based on non-symmetric left window functions are*

$$\frac{d}{dx}Lf(x_k) = \frac{f_k - f_{k-1}}{h} - \frac{2(f_k - 2f_{k-1} + f_{k-2})}{h(4+\xi)} \quad \textit{for } R = 3h, \tag{3.6}$$

where $\xi := \frac{w_{k-1}}{w_{k-2}}$.

Proof. Let $R = 3h$, the denominator of the right hand side of (3.4) and (3.5) is given as

$$x_k^2 \sum_{j,j\neq k} w_j - 2x_k \sum_{j,j\neq k} w_j x_j + \sum_{j,j\neq k} w_j x_j^2 = h^2(w_{k-1} + 4w_{k-2}).$$

For $j = k$, (3.4) reads as

$$\frac{d}{dx}\phi_k(x_k) = \frac{h(w_{k-1} + 2w_{k-2})}{h^2(w_{k-1} + 4w_{k-2})} = \frac{\xi + 2}{h(\xi + 4)}, \tag{3.7}$$

for $j \neq k$, we have

$$\frac{d}{dx}\phi_{k-1}(x_k) = \frac{-hw_{k-1}}{h^2(w_{k-1} + 4w_{k-2})} = \frac{-\xi}{h(\xi + 4)} \quad \text{and} \tag{3.8}$$

$$\frac{d}{dx}\phi_{k-2}(x_k) = \frac{-2hw_{k-2}}{h^2(w_{k-1} + 4w_{k-2})} = \frac{-2}{h(\xi + 4)}. \tag{3.9}$$

Substituting (3.7), (3.8), (3.9) to (3.3), we obtain (3.6). ■

Lemma 3.2. *In case of symmetric window functions, the difference formulae are given by*

$$\frac{d}{dx}Lf(x_k) = \frac{f_{k+1} - f_{k-1}}{2h} \quad \textit{for } R = 2h, \tag{3.10}$$

and

$$\frac{d}{dx}Lf(x_k) = \frac{f_{k+1} - f_{k-1}}{2h} + \frac{(f_{k+2} - 2f_{k+1} + 2f_{k-1} - f_{k-2})}{h(4+\xi)} \quad \textit{for } R = 3h. \tag{3.11}$$

For more information of the formulae (3.6), (3.10), and (3.11), we use Taylor series expansion

$$f(x_k) = f(x_j) + hf'(x_j)(x_j - x_k) + \frac{1}{2!}f''(x_j)(x_j - x_k)^2 + \mathcal{O}(h^3)$$

which we introduce into equation (3.6) (3.10), and (3.11), and obtain the following

Lemma 3.3. *The difference formulae are the approximations to the first derivative of $f(x)$. In particular,* (3.6) *gives*

$$\frac{d}{dx}Lf(x_k) = f'(x_k) - \frac{2hf''(x_k)}{4+\xi}. \tag{3.12}$$

(3.10) *leads to*

$$\frac{d}{dx}Lf(x_k) = f'(x_k) + \mathcal{O}(h^2). \tag{3.13}$$

(3.11) *gives*

$$\frac{d}{dx}Lf(x_k) = f'(x_k) + \frac{2h^2f'''(x_k)}{4+\xi}. \tag{3.14}$$

In the symmetric case with the specific parameter $R = 2h$, the method is of second order.

3.1.2 Quadratic basis

The Gram matrix $\tilde{G}$ in equation (2.33) reads as

$$\tilde{G}(x) := \begin{pmatrix} a & b & d \\ b & d & f \\ d & f & g \end{pmatrix} = \begin{pmatrix} \sum_{\substack{j=1 \\ j\neq i}}^{N} w_j & \sum_{\substack{j=1 \\ j\neq i}}^{N} w_j x_j & \sum_{\substack{j=1 \\ j\neq i}}^{N} w_j x_j^2 \\ \sum_{\substack{j=1 \\ j\neq i}}^{N} w_j x_j & \sum_{\substack{j=1 \\ j\neq i}}^{N} w_j x_j^2 & \sum_{\substack{j=1 \\ j\neq i}}^{N} w_j x_j^3 \\ \sum_{\substack{j=1 \\ j\neq i}}^{N} w_j x_j^2 & \sum_{\substack{j=1 \\ j\neq i}}^{N} w_j x_j^3 & \sum_{\substack{j=1 \\ j\neq i}}^{N} w_j x_j^4 \end{pmatrix} \tag{3.15}$$

and its inverse is

$$\tilde{G}(x)^{-1} = \frac{1}{\psi} \cdot \begin{pmatrix} -f^2 + dg & df - bg & -d^2 + bf \\ df - bg & -d^2 + ag & bd - af \\ -d^2 + bf & bd - af & -b^2 + ad \end{pmatrix} \tag{3.16}$$

where $\psi := -d^3 + 2bdf - af^2 - b^2g + adg$. The first derivatives of kernel functions are given by

$$\frac{d}{dx}\phi_i(x_i) = \frac{\tau}{\zeta} \tag{3.17}$$

$$\frac{d}{dx}\phi_j(x_i) = \frac{\omega}{\zeta}, \quad j \neq i, \tag{3.18}$$

where

$$\tau = -df + bg + (3d^2 - 2bf - ag)x_i + (3af - 3bd)x_i^2 + 2(b^2 - ad)x_i^3 \tag{3.19}$$

$$\omega = w_j(x_i - x_j)(g - 3fx_i + 3dx_i^2 - bx_i^3 - fx_j + 3dx_ix_j \tag{3.20}$$

$$- 3bx_i^2x_j + ax_i^3x_j \tag{3.21}$$

$$\zeta = f^2 - dg + 2(bg - df)x_i + (3d^2 - 2bf - ag)x_i^2 \tag{3.22}$$

$$+ 2(af - bd)x_i^3 + (b^2 - ad)x_i^4 \tag{3.23}$$

Lemma 3.4. *Let $R = 4h$, an explicit expression of the first derivative from IMLS method based on non-symmetric window functions is*

$$\begin{aligned}\frac{d}{dx}Lf(x_k) = &\frac{3f_k - 4f_{k-1} + f_{k-2}}{2h} \\ &+ \frac{3(\xi+4)(f_{k-3} - 3f_{k-2} + 3f_{k-1} - f_k)}{2h(9 + 9\xi + \xi_1)}\end{aligned} \tag{3.24}$$

where $\xi = \frac{w_{k-1}}{w_{k-2}}$ and $\xi_1 = \frac{w_{k-1}}{w_{k-3}}$.

Proof. For $R = 4h$, the expression of the denominator of (3.17), (3.18) reads as

$$\zeta = 4h^6(w_{k-1}w_{k-2} + 9w_{k-1}w_{k-3} + 9w_{k-2}w_{k-3}). \tag{3.25}$$

The first derivative of the IMLS shape function at point $x_j = x_k$ is given as

$$\begin{aligned}\frac{d}{dx}\phi_k(x_k) &= \frac{6h^5(w_{k-1}w_{k-2} + 8w_{k-1}w_{k-3} + 5w_{k-2}w_{k-3})}{4h^6(w_{k-1}w_{k-2} + 9w_{k-1}w_{k-3} + 9w_{k-2}w_{k-3})} \\ &= \frac{6(\xi_1 + 8\xi + 5)}{4h(\xi_1 + 9\xi + 9)}.\end{aligned} \tag{3.26}$$

For $x_j \neq x_k$, we have

$$\frac{d}{dx}\phi_{k-1}(x_k) = \frac{-2(4\xi_1 + 27\xi)}{4h(\xi_1 + 9\xi + 9)} \tag{3.27}$$

$$\frac{d}{dx}\phi_{k-2}(x_k) = \frac{2(\xi_1 - 27)}{4h(\xi_1 + 9\xi + 9)} \tag{3.28}$$

$$\frac{d}{dx}\phi_{k-3}(x_k) = \frac{6(\xi + 4)}{4h(\xi_1 + 9\xi + 9)}. \tag{3.29}$$

Substituting (3.26) - (3.29) to (3.3), we obtain (3.24). ■

Lemma 3.5. *In the symmetric window function cases, the difference formulae of the first derivatives of IMLS interpolants are*

$$\frac{d}{dx}Lf(x_k) = \frac{f_{k+1} - f_{k-1}}{2h} + \frac{(f_{k+2} - 2f_{k+1} + 2f_{k-1} - f_{k-2})}{h(4+\xi)} \tag{3.30}$$

for $R = 3h$ *and*

$$\begin{aligned}\frac{d}{dx}Lf(x_k) = \frac{f_{k+1} - f_{k-1}}{2h} + \frac{2\xi(f_{k+2} - 2f_{k+1} + 2f_{k-1} - f_{k-2})}{2h(1+4\xi+9\xi_1)} \\ + \frac{3\xi_1(f_{k+3} - 3f_{k+1} + 3f_{k-1} - f_{k-3})}{2h(1+4\xi+9\xi_1)}\end{aligned} \tag{3.31}$$

for $R = 4h$.

Lemma 3.6. *The difference formulae* (3.24),(3.30), *and* (3.31) *are the approximations to the first derivative of* $f(x)$. (3.24) *leads to*

$$\frac{d}{dx}Lf(x_k) = f'(x_k) + \frac{3h^2(\xi+4)f'''(x_k)}{2(9+9\xi+\xi_1)}. \tag{3.32}$$

(3.30) *provides*

$$\frac{d}{dx}Lf(x_k) = f'(x_k) + \frac{2h^2 f'''(x_k)}{4+\xi}, \tag{3.33}$$

and (3.31) *gives*

$$\frac{d}{dx}Lf(x_k) = f'(x_k) - \frac{h^2\xi f'''(x_k)}{1+4\xi+9\xi_1} + \frac{12h^2\xi_1 f'''(x_k)}{1+4\xi+9\xi_1}. \tag{3.34}$$

Proof. Use Taylor expansions as in Lemma 3.3. ■

In the next section we will derive the explicit form of the second derivatives of IMLS interpolants.

3.2 Difference formulae of the second derivatives of interpolants

For the second derivatives, we will consider only the method based on a quadratic basis. In general, the second derivatives do not exist for the method based on linear basis.

Theorem 3.7. *The second derivatives of our IMLS method based on a linear basis is inconsistent.*

Proof. The proof is given by Sonar [Son05]. ■

We now turn back to the quadratic case. According to the expression of the derivatives of IMLS shape functions in Chapter 2 Section 2.4, the second derivative can be written as

$$\frac{d^2}{dx^2}\phi_i(x_i) = \frac{1}{p^T(x_i)\tilde{G}(x)p(x_i)} p^T(x_i)\tilde{G}(x_i)^{-1}\frac{d^2}{dx^2}p(x_i), \tag{3.35}$$

$$\frac{d^2}{dx^2}\phi_j(x_i) = w_j(x_i)p^T(x_j)\left[\tilde{G}(x_i)^{-1} - \frac{\left(\tilde{G}(x_i)^{-1}p(x_i)\right)\left(\tilde{G}(x_i)p(x_i)\right)^T}{p^T(x_i)\tilde{G}(x_i)p(x_i)}\right]\frac{d^2}{dx^2}p(x_i),\ j \neq i. \tag{3.36}$$

Another form of the second derivatives reads as

$$\frac{d^2}{dx^2}\phi_i(x_i) = \frac{2(bf - d^2 + x_i(bd - af) + x_i^2(ad - f^2))}{\zeta} \tag{3.37}$$

$$\frac{d^2}{dx^2}\phi_j(x_i) = \frac{2w_j\eta}{\zeta}, \quad j \neq i, \tag{3.38}$$

where

$$\eta := x_j^2(d - 2bx_i + ax_i^2) + x_j(-f + dx_i + bx_i^2 - ax_i^3) + fx_i - 2dx_i^2 + bx_i^3 \tag{3.39}$$

and ζ is defined by (3.23). The constants a, b, d, and f are defined in (3.15). The second derivative of the function $Lf(x)$ is

$$\frac{d^2}{dx^2}Lf(x) = \sum_{j=1}^{N}\frac{d^2}{dx^2}\phi_j(x)f_j. \tag{3.40}$$

The detailed information on the second derivative of IMLS interpolants is summerized as follows

Lemma 3.8. *Based on the non-symmetric window function with the fixed parameter $R = 4h$, the difference formula of IMLS interpolants is given as*

$$\begin{aligned}\frac{d^2}{dx^2}Lf(x_k) = &\frac{f_k - 2f_{k-1} + f_{k-2}}{h^2} \\ &- \frac{3(\xi + 2)(f_k - 3f_{k-1} + 3f_{k-2} - f_{k-3})}{h^2(9 + 9\xi + \xi_1)}\end{aligned} \tag{3.41}$$

where $\xi = \frac{w_{k-1}}{w_{k-2}}$ and $\xi_1 = \frac{w_{k-1}}{w_{k-3}}$.

Proof. For the fixed parameter $R = 4h$, the second derivatives of the IMLS shape

functions are summerized as

$$\begin{aligned}\frac{d^2}{dx^2}\phi_k(x_k) &= \frac{4h^4(w_{k-1}w_{k-2}+6w_{k-1}w_{k-3}+3w_{k-2}w_{k-3})}{4h^6(w_{k-1}w_{k-2}+9w_{k-1}w_{k-3}+9w_{k-2}w_{k-3})}\\ &= \frac{\xi_1+6\xi+3}{h^2(\xi_1+9\xi+9)}\end{aligned} \tag{3.42}$$

$$\begin{aligned}\frac{d^2}{dx^2}\phi_{k-1}(x_k) &= \frac{4h^4(-2w_{k-1}w_{k-2}-9w_{k-1}w_{k-3})}{4h^6(w_{k-1}w_{k-2}+9w_{k-1}w_{k-3}+9w_{k-2}w_{k-3})}\\ &= \frac{-2\xi_1-9\xi}{h^2(\xi_1+9\xi+9)}\end{aligned} \tag{3.43}$$

$$\begin{aligned}\frac{d^2}{dx^2}\phi_{k-2}(x_k) &= \frac{4h^4(w_{k-1}w_{k-2}-9w_{k-2}w_{k-3})}{4h^6(w_{k-1}w_{k-2}+9w_{k-1}w_{k-3}+9w_{k-2}w_{k-3})}\\ &= \frac{\xi_1-9}{h^2(\xi_1+9\xi+9)}\end{aligned} \tag{3.44}$$

$$\begin{aligned}\frac{d^2}{dx^2}\phi_{k-3}(x_k) &= \frac{12h^4(w_{k-1}w_{k-3}+2w_{k-2}w_{k-3})}{4h^6(w_{k-1}w_{k-2}+9w_{k-1}w_{k-3}+9w_{k-2}w_{k-3})}\\ &= \frac{3(\xi+2)}{h^2(\xi_1+9\xi+9)}.\end{aligned} \tag{3.45}$$

Substituting the second derivatives of IMLS shape functions from (3.42) - (3.45) to (3.40), the second derivatives of the IMLS interpolants read as

$$\begin{aligned}\frac{d^2}{dx^2}Lf(x) &= \sum_{j=1}^{N}\frac{d^2}{dx^2}\phi_j(x)f_j\\ &= \frac{\xi_1(f_k-2f_{k-1}+f_{k-2})+3\xi(2f_k-3f_{k-1}+f_{k-3})+3(f_k-3f_{k-2}+2f_{k-3})}{h^2(\xi_1+9\xi+9)}\\ &= \frac{f_k-2f_{k-1}+f_{k-2}}{h^2}-\frac{3(\xi+2)(f_k-3f_{k-1}+3f_{k-2}-f_{k-3})}{h^2(\xi_1+9\xi+9)}.\end{aligned}$$

■

Lemma 3.9. *The approximations of the second derivatives of IMLS interpolants based on symmetric window function are*

$$\begin{aligned}\frac{d^2}{dx^2}Lf(x_k) &= \frac{f_{k+1}-2f_k+f_{k-1}}{h^2}\\ &\quad+\frac{4(f_{k+2}-4f_{k+1}+6f_k-4f_{k-1}+f_{k-2})}{h^2(16+\xi)}\end{aligned} \tag{3.46}$$

for the specific parameter $R = 3h$*, and*

$$\begin{aligned}\frac{d^2}{dx^2}Lf(x_k) = &\frac{f_{k+1} - 2f_k + f_{k-1}}{h^2} \\ &+ \frac{4\xi(f_{k+2} - 4f_{k+1} + 6f_k - 4f_{k-1} + f_{k-2})}{h^2(1 + 16\xi + 81\xi_1)} \\ &+ \frac{9\xi_1(f_{k+3} - 9f_{k+1} + 16f_k - 9f_{k-1} + f_{k-3})}{h^2(1 + 16\xi + 81\xi_1)}\end{aligned} \tag{3.47}$$

for $R = 4h$*.*

Lemma 3.10. *The difference formulae are approximations to the second derivative of* $f(x)$*. In particular,* (3.41) *gives*

$$\frac{d^2}{dx^2}Lf(x_k) = f''(x_k) - \frac{3h(\xi + 2)f'''(x_k)}{9 + 9\xi + 9\xi_1}. \tag{3.48}$$

(3.46) *yields*

$$\frac{d^2}{dx^2}Lf(x_k) = f''(x_k) + \frac{4h^2 f^{(4)}(x_k)}{16 + \xi}, \tag{3.49}$$

and (3.47) *gives*

$$\frac{d^2}{dx^2}Lf(x_k) = f''(x_k) + \frac{4h^2 f^{(4)}(x_k)}{1 + 16\xi + 81\xi_1} + \frac{9\xi_1 h^{(6)} f^{(6)}(x_k)}{1 + 16\xi + 81\xi_1} \tag{3.50}$$

Proof. Use Taylor expansions in (3.41), (3.46), (3.47). ■

At the end of this chapter, we conclude the difference formulae from IMLS methods in Table 3.1 for the first derivatives and in Table 3.2 for the second derivatives which are compared with classical upwind and central differences.

TABLE 3.1
Difference formulae for the first derivatives

basis	non-symmetric weight	symmetric weight	FDM
• linear			
$R = 2h$	$-$	$\tilde{f}'_f(x_k) = \frac{f_{k+1}-f_{k-1}}{2h}$	$f'(x_k) = \frac{f_{k+1}-f_{k-1}}{2h}$
$R = 3h$	$\tilde{f}'_l(x_k) = \frac{f_k-f_{k-1}}{h}$ $-\frac{2(f_k-2f_{k-1}+f_{k-2})}{h(4+\xi)}$	$\tilde{f}'_f(x_k) = \frac{f_{k+1}-f_{k-1}}{2h}$ $+\frac{f_{k+2}-2f_{k+1}+2f_{k-1}-f_{k-2}}{h(4+\xi)}$	$f'(x_k) = \frac{f_k-f_{k-1}}{h}$
• quadratic			
$R = 3h$	$-$	$\tilde{f}'_f(x_k) = \frac{f_{k+1}-f_{k-1}}{2h}$ $+\frac{f_{k+2}-2f_{k+1}+2f_{k-1}-f_{k-2}}{h(4+\xi)}$	$f'(x_k) = \frac{f_{k+1}-f_{k-1}}{2h}$
$R = 4h$	$\tilde{f}'_l(x_k) = \frac{3f_k-4f_{k-1}+f_{k-2}}{2h}$ $+\frac{3(\xi+4)(f_{k-3}-3f_{k-2}+3f_{k-1}-f_k)}{2h(9+9\xi+\xi_1)}$	$\tilde{f}'_f(x_k) = \frac{f_{k+1}-f_{k-1}}{2h}$ $+\frac{2\xi(f_{k+2}-2f_{k+1}+2f_{k-1}-f_{k-2})}{h(9+9\xi+\xi_1)}$	$f'(x_k) = \frac{3f_k-4f_{k-1}+f_{k-2}}{2h}$

TABLE 3.2
Difference formulae for the second derivatives

basis	non-symmetric weight	symmetric weight	FDM
• quadratic $R = 3h$	–	$\tilde{f}''_f(x_k) = \frac{(f_{k+1}-2f_k+f_{k-1})}{h^2}$ $+\frac{4(f_{k+2}-4f_{k+1}+6f_k-4f_{k-1}+f_{k-2})}{h^2(16+\xi)}$	$f''(x_k) = \frac{f_{k+1}-2f_k+f_{k-1}}{h^2}$
$R = 4h$	$\tilde{f}''_l(x_k) = \frac{(f_k-2f_{k-1}+f_{k-2})}{h^2}$ $-\frac{3(\xi+2)(f_k-3f_{k-1}+3f_{k-2}-f_{k-3})}{h^2(9+9\xi+\xi_1)}$	$\tilde{f}''_f(x_k) = \frac{(f_{k+1}-2f_k+f_{k-1})}{h^2}$ $+\frac{4\xi(f_{k+2}-4f_{k+1}+6f_k-4f_{k-1}+f_{k-2})}{h^2(1+16\xi+81\xi_1)}$ $+\frac{9\xi_1(f_{k+3}-9f_{k+1}+16f_k-9f_{k-1}+f_{k-3})}{h^2(1+16\xi+81\xi_1)}$	$f''(x_k) = \frac{f_k-2f_{k-1}+f_{k-2}}{h^2}$ $f''(x_k) = \frac{f_{k+1}-2f_k+f_{k-1}}{h^2}$

Chapter 4

Application of IMLS method

In this chapter, we will present some numerical computations with our IMLS methods. For tests on accuracy the derivatives of IMLS interpolants will be examined by choosing Franke's function as test functions. Later, an application to a one-dimensional linear advection equation with smooth initial data will be disscussed.

4.1 Accuracy tests

As a standard test, we select Franke's function which is often used in the literature for the method of interpolation. Franke's function is initially defined on the interval $[-1, 1]$ in one dimension and is defined as follows

$$f(x) = \frac{3}{4}e^{-\frac{(9x-2)^2}{4}} + \frac{3}{4}e^{-\frac{9x+1)^2}{49}} + \frac{1}{2}e^{-\frac{(9x-7)^2}{4}} - \frac{1}{5}e^{-(9x-4)^2}. \tag{4.1}$$

Figure 4.1 show Franke's function and its first and second derivatives. The discrete norms of the error are given by

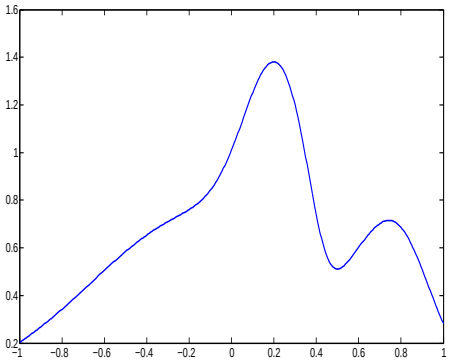
$$\begin{aligned} L_1\ error &= h\sum_{j=1}^{N}|f(x_j) - \tilde{f}(x_j)| \\ L_2\ error &= \sqrt{h\sum_{j=1}^{N}|f(x_j) - \tilde{f}(x_j)|^2} \\ L_\infty\ error &= \max_{1\le j\le N}|f(x_j) - \tilde{f}(x_j)| \end{aligned} \tag{4.2}$$

and an estimate of the order of convergence is computed by

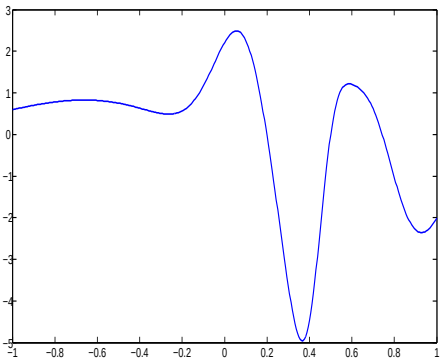
$$\frac{\ln(err(h_1)/err(h_2))}{\ln(h_1/h_2)} \tag{4.3}$$

where h_1 and h_2 are any two grid spacings.

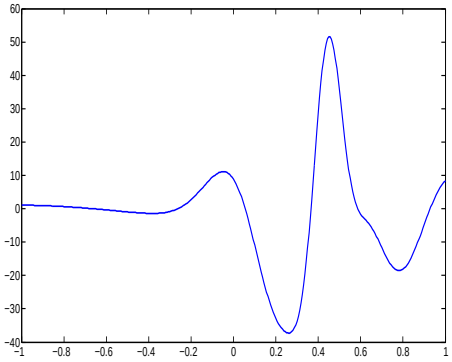
Table 4.1 and Table 4.2 show the results obtained by our IMLS method based on a linear basis. It can be seen that the results obtained by using a non-symmetric



(a) Franke function



(b) The first derivative



(c) The second derivative

FIGURE 4.1 *Franke's function and its derivatives.*

window function for a fix $R = 3h$ in Table 4.1 are first order accurate in the discrete L_1, L_2, and L_∞ norms. Comparing these results with the solutions computed by applying a symmetric window function ($R = 2h$) as shown in Table 4.2, we see that the method based on symmetric window functions produces smaller errors and gives better accuracy.

TABLE 4.1
Linear basis ($R = 3h$): $||f'(x) - \tilde{f}'_l(x)||$ and order of accuracy

N	L_1 error	L_1 order	L_2 error	L_2 order	L_∞ error	L_∞ order
20	4.63E-001	-	6.98E-001	-	1.75E+000	-
40	2.28E-001	1.02	3.56E-001	0.97	9.16E-001	0.94
80	1.16E-001	0.97	1.79E-001	1.00	4.67E-001	0.97
160	5.82E-002	1.00	8.93E-002	1.00	2.32E-001	1.01
320	2.91E-002	0.99	4.45E-002	0.99	1.16E-001	0.99

TABLE 4.2
Linear basis ($R = 2h$): $||f'(x) - \tilde{f}'_f(x)||$ and order of accuracy

N	L_1 error	L_1 order	L_2 error	L_2 order	L_∞ error	L_∞ order
20	1.62E-001	-	2.66E-001	-	8.75E-001	-
40	4.47E-002	1.85	6.90E-002	1.94	2.35E-001	1.90
80	1.11E-002	2.01	1.73E-002	2.00	5.85E-002	2.01
160	1.77E-003	1.99	4.30E-003	2.01	1.48E-002	1.98
320	2.89E-004	2.00	1.10E-003	1.94	3.70E-003	1.98

In case of a quadratic basis with a specific dilation parameter $R = 3h$, the magnitudes of errors and numerical orders of accuracy are shown in Table 4.3 for the first derivative test study and in Table 4.4 for the second derivative test study.

The experimental orders of accuracy in Table 4.3 are of second order which are the same order as in Table 4.2. We observe that magnitudes of the errors in Table 4.2 from the methods based on a linear basis are smaller because this method achieves the optimal second order and is exactly the second order central difference, see Lemma 3.2 in (3.10).

TABLE 4.3
Quadratic basis ($R = 3h$): $||f'(x) - \tilde{f}'_f(x)||$ and order of accuracy

N	L_1 error	L_1 order	L_2 error	L_2 order	L_∞ error	L_∞ order
20	2.18E-001	-	3.47E-001	-	1.08E+000	-
40	5.44E-002	2.00	9.51E-002	1.87	3.30E-001	1.71
80	1.41E-002	1.95	2.47E-002	1.94	9.42E-002	1.81
160	3.60E-003	1.97	6.30E-003	1.97	2.40E-002	1.97
320	8.92E-004	1.99	1.60E-003	1.95	6.00E-003	1.98

In Table 4.4, we list the discrete L_1, L_2, and L_∞ errors and their accuracy for the approximation of the second derivatives. The numerical results confirm an expected

order of accuracy as in Lemma 3.10.

TABLE 4.4
Quadratic basis($R = 3h$): $||f''(x) - \tilde{f}_f''(x)||$ and order of accuracy

N	L_1 error	L_1 order	L_2 error	L_2 order	L_∞ error	L_∞ order
20	1.62E+000	-	2.72E+000	-	7.40E+000	-
40	5.32E-001	1.61	1.03E+000	1.40	4.59E+000	0.69
80	1.39E-002	1.94	2.76E-001	1.90	1.25E+000	1.88
160	3.51E-002	1.98	7.02E-002	1.97	3.19E-001	1.97

4.2 IMLS method to linear advection equation

The following description is mainly based on [Hir90].

4.2.1 Monotone schemes

The general form of a numerical scheme applied to the scalar conservation equation $u_t + f(u)_x = 0$ can be written as

$$u_i^{n+1} = H(u_{i-k}^n, u_{i-k+1}^n, \ldots, u_{i+k}^n). \tag{4.4}$$

Definition 2. *The numerical scheme*(4.4) *is called* ***monotone*** *if H is a monotonically increasing function of each of its arguments, that is*

$$\frac{\partial H}{\partial u_j}(u_{i-k}, u_{i-k+1}, \ldots, u_{i+k}) \geq 0 \quad \textit{for all} \quad i-k \leq j \leq i+k. \tag{4.5}$$

For a general linear case,

$$v_i^{n+1} = \sum_k b_k v_{i+k}^n, \tag{4.6}$$

the condition of monotonicity is given by

$$b_k \geq 0. \tag{4.7}$$

As in the literature [Hir90, LeV99], monotone schemes are at most of the first order accurate. This limitation is not useful for practical numerical tools. Then a weaker condition than monotonicity, has to be defined by providing the idea of *total variation*.

4.2.2 Total variation

The total variation (TV) of a function $u(x)$ is given by

$$TV(u) = \int \left| \frac{\partial u}{\partial x} dx \right|. \tag{4.8}$$

A function $u(x,t)$ is said to be of bounded variation, if $TV(u)$ stays bounded for all times t. For the discrete version of the solution the total variation is defined by

$$TV(u) = \sum_i |u_{i+1} - u_i| \tag{4.9}$$

and TV should be uniformly bounded in t and h. Moreover, a numerical scheme is called *total variation dimishing* (TVD) if

$$TV(u^{n+1}) \leq TV(u^n) \text{ for all } n. \tag{4.10}$$

The numerical experiments on the TV property for hyperbolic conservation laws will be presented in Chapter 7.

Let us first discuss the linear advection equation

$$u_t + u_x = 0. \tag{4.11}$$

Lemma 4.1. *If we discretize* (4.11) *in time by using the first order Euler scheme and in space by applying IMLS method based on the linear basis and non-symmetric window functions with the specific dilation parameter $R = 3h$, then the method is monotone.*

Proof. Euler time integration reads as

$$u_i^{n+1} = u_i^n - \Delta t (u_x)_i^n. \tag{4.12}$$

The second term of the right hand side (4.12) is

$$(u_x)_i^n = \sum_{j=1}^{N} \frac{d}{dx} \phi_j^l(x_i) u_j.$$

By the help of Lemma 3.6, equation (4.12) can be rewritten as

$$\begin{aligned} u_i^{n+1} &= \left\{ 1 - \lambda + \frac{2\lambda}{4+\xi} \right\} u_i^n + \left\{ \lambda - \frac{4\lambda}{4+\xi} \right\} u_{i-1}^n + \frac{2\lambda}{4+\xi} u_{i-2}^n \\ &= \left\{ 1 - \frac{\lambda(2+\xi)}{4+\xi} \right\} u_i^n + \left\{ \frac{\lambda\xi}{4+\xi} \right\} u_{i-1}^n + \frac{2\lambda}{4+\xi} u_{i-2}^n, \end{aligned}$$

where $\lambda = \frac{\Delta t}{h}$. It is seen that for $0 < \lambda \leq 1$, all of the coefficients of u at the right side of the above equation are positive. Therefore, the scheme is monotone. ■

4.2.3 Lax-Wendroff method

Since the above scheme is monotone, it is only of the first order. For more accurate results, we discretize in time by the Lax-Wendroff (LW) method. The method reads as

$$u_i^{n+1} = u_i^n - \Delta t(u_x)_i^n + \frac{\Delta t^2}{2}(u_{xx})_i^n. \tag{4.13}$$

The particular species of IMLS methods will be divided into two cases depending on the types of window functions used corresponding to the local domain of computation, see Figure 4.2.

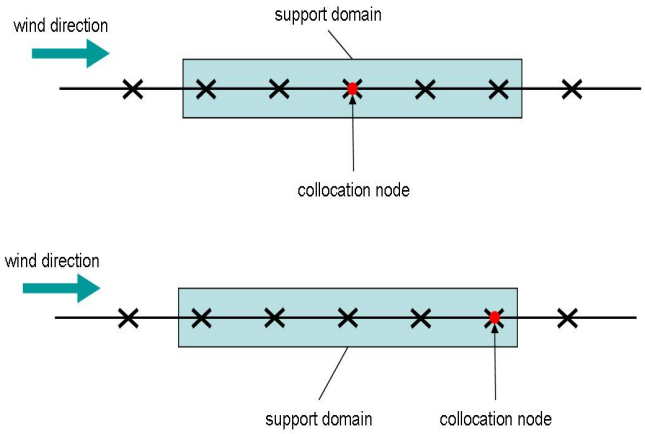


FIGURE 4.2 *Different local domains of interest.*

- **IQS**: IMLS method based on quadratic basis and symmetric window function with the fixed dilation parameter $R = 3h$.
- **IQNS**: IMLS method based on quadratic basis and non-symmetric left window function with $R = 4h$.

4.3 Von Neumann stability analysis

In this section the method to find the stability condition for a numerical scheme will be presented. The following description is based on [Tra09].
We are interested only in the linear advection equation (4.11). The fourier mode of the solution of the linear advection equation satisfies

$$u_j^n = G(k)^n e^{ik(jh)} \tag{4.14}$$

where $G(k) = e^{-ak}$ and k is the *wave number*. Let us define the numerical ratio by

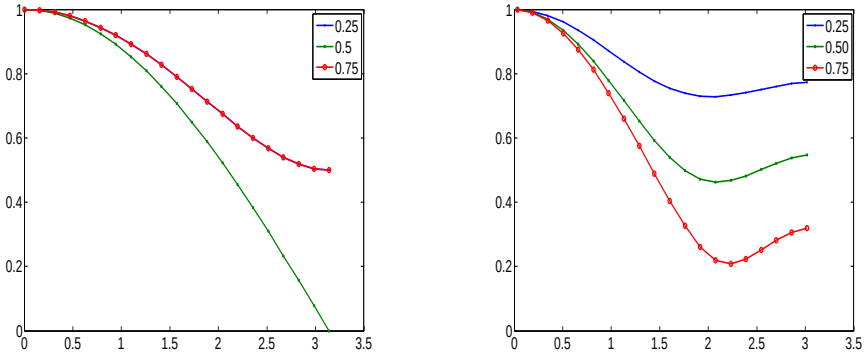
$$G(k) = \frac{u^{n+1}}{u^n}.$$

The numerical scheme is *dissipative* if and only if $|G| < 1$ for all $\theta \neq 0$ where $\theta = kh$ and *dispersive* if and only if $\mathsf{arg}(G)/\theta$ depends on θ. Note that any complex number z has the polar form $z = |z|e^{i\upsilon}$ with $\mathsf{arg}(z) = \upsilon$. The total *numerical dissipation (amplitude) error* is $1 - |G|^{1/\lambda}$. The total numerical dispersion is measured by the *phase error* and given as $1 + \mathsf{arg}(G)^{1/\lambda}/\theta$. See [ATP84, MM94, Tho99, Tra09] for more details.

The dissipation error measures how much the amplitude of that information changes, while the phase error measures how fast the information moves. Positive phase errors show that the information is moving slower than it should. Negative dissipation errors indicate the unstable solution because the amplitude is larger than it should be.

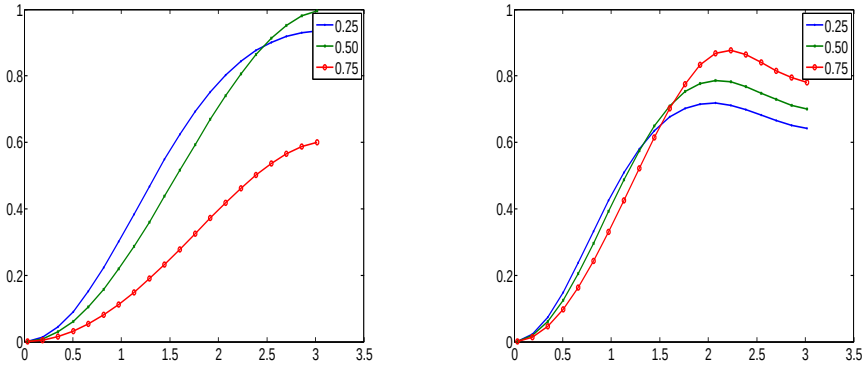
The amplification factors of Fourier modes of the methods are obtained by inserting the Ansatz (4.14) into the discrete version of the linear advection equation. In this section we consider the explicit Euler method and the Lax-Wendroff method for the behaviour of Fourier modes. We show the results over the range $0 \leq kh \leq \pi$ for the amplification factors and the dissipiation errors. The results of the phases errors is over the inteval $0 \leq kh \leq \frac{\pi}{2}$. Figure 4.3 shows graphs of the amplification factors for the explicit upwind scheme, the components of these factors are explicitly given in Table 4.5, Figure 4.4 shows graphs of the dissipation errors, and Figure 4.5 shows the phase errors. We also display the results of a Fourier analysis of the Lax-Wendroff scheme in Figure 4.6 for the amplification factors, in Figure 4.7 for the dissipation errors, and in Figure 4.8 for the phase errors. We show the components of the amplification factors for the Lax-Wendroff scheme in Table 4.6.

The modulus of the amplification factors, $|G|$, particularly decides the stability of the scheme. In order for the schemes to be stable, the modulus of this factor must be less than or equal to one, $|G| \leq 1$ for all possible values of k. As we display in Figure 4.3 for upwind methods, both schemes - upwind FDM (see Figure 4.3(a)) and upwind IMLS (see Figure 4.3(b)) defined in Lemma 4.1 - have the modulus of $|G|$ less than or equal to one for $\lambda = 0.25, 0, 50, 0, 75$. For the Lax-Wendroff scheme the modulus of $|G|$ of the LW-FDM, the LW-IQS, and the LW-IQNS are also less than one, see Figure 4.6. These quatities are close to unity when kh is small for the LW-FDM scheme and the LW-IQNS scheme, see Figure 4.6(a), 4.6(c). In Figure 4.8(c) it shows that in case of the LW-IQNS scheme the Fourier component of



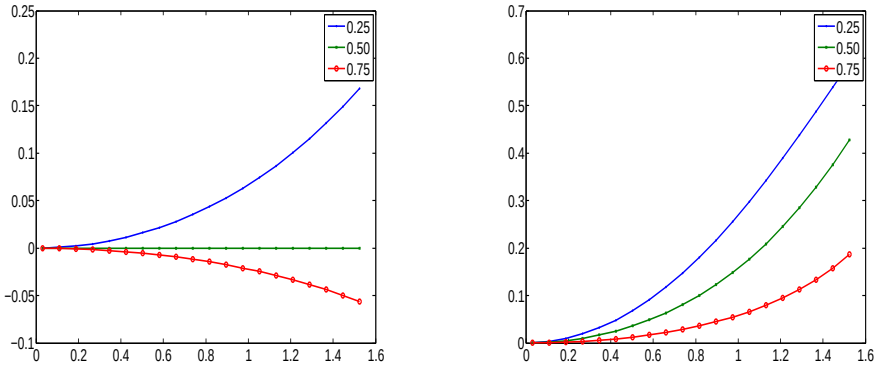
(a) Upwind difference (b) Upwind-IMLS

FIGURE 4.3 *Amplification factors of Fourier modes for explicit Euler time discretization, plot against kh, for* $\lambda = \frac{\Delta t}{h} = 0.25, 0.50, 0.75$



(a) Upwind difference (b) Upwind-IMLS

FIGURE 4.4 *Dissipation errors for explicit Euler time discretization, plot against kh, for* $\lambda = \frac{\Delta t}{h} = 0.25, 0.50, 0.75$



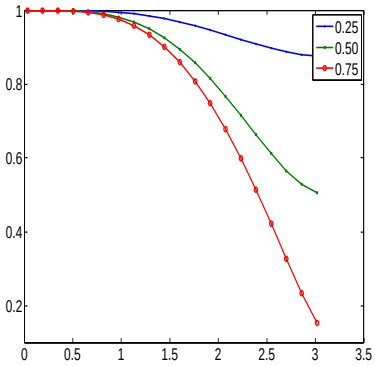
(a) Upwind difference (b) Upwind-IMLS

FIGURE 4.5 *Phase errors for explicit Euler time discretization, plot against kh, for* $\lambda = \frac{\Delta t}{h} = 0.25, 0.50, 0.75$

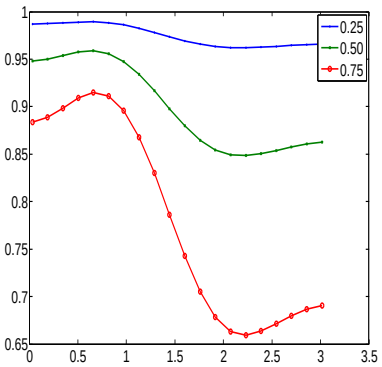
TABLE 4.5
Components of amplification factors of Fourier modes for the explicit Euler scheme

Method	Upwind-FDM	Upwind-IMLS
Real parts	$1-\lambda(1-\cos(kh))$	$1-\lambda(1-\cos(kh))$ $+\frac{2\lambda}{4+\xi}{}^{a}(1-2\cos(kh)+\cos(2kh))$
Imaginary parts	$-\lambda\sin(kh)$	$-\lambda\sin(kh)+\frac{2\lambda}{4+\xi}(2\sin(kh)+\cos(2kh))$

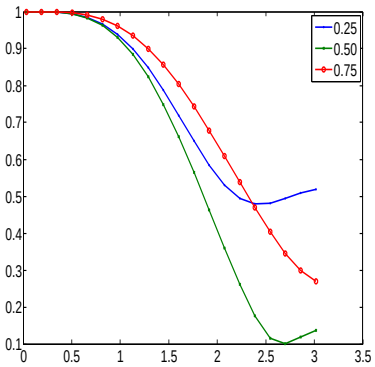
[a] $\xi = \frac{w_{k-1}}{w_{k-2}}, R = 3h$



(a) LW-FDM

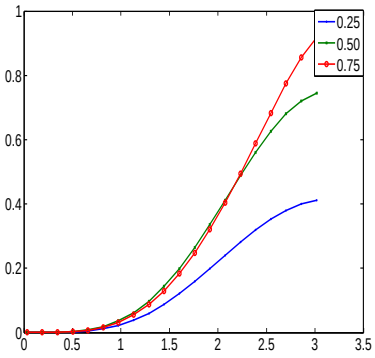


(b) LW-IQS

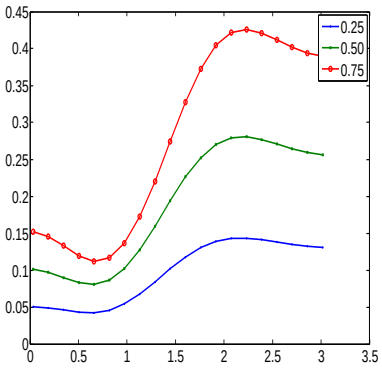


(c) LW-IQNS

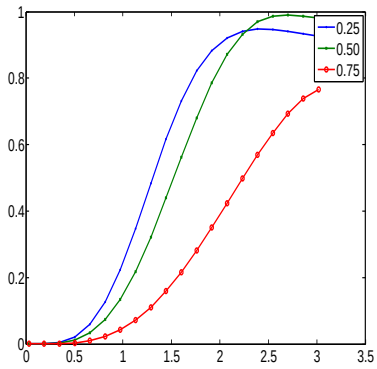
FIGURE 4.6 *Amplification factors of Fourier modes for Lax-Wendroff time discretization, plot against kh, for* $\lambda = \frac{\Delta t}{h} = 0.25, 0.50, 0.75$



(a) LW-FDM

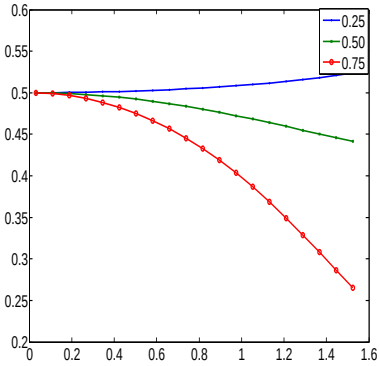


(b) LW-IQS

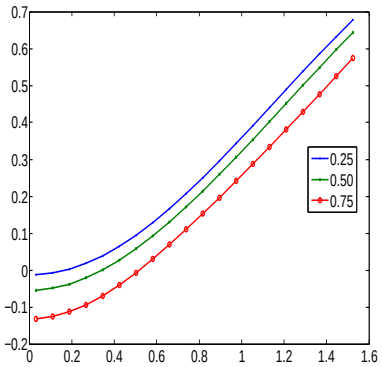


(c) LW-IQNS

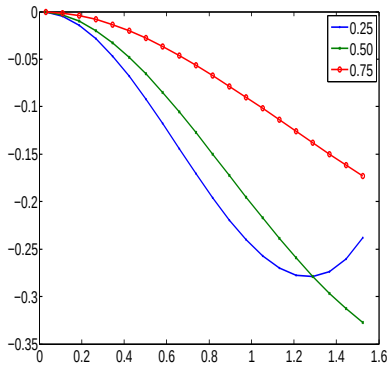
FIGURE 4.7 *Dissipation errors for Lax-Wendroff time discretization, plot against* kh*, for* $\lambda = \frac{\Delta t}{h} = 0.25, 0.50, 0.75$



(a) LW-FDM



(b) LW-IQS



(c) LW-IQNS

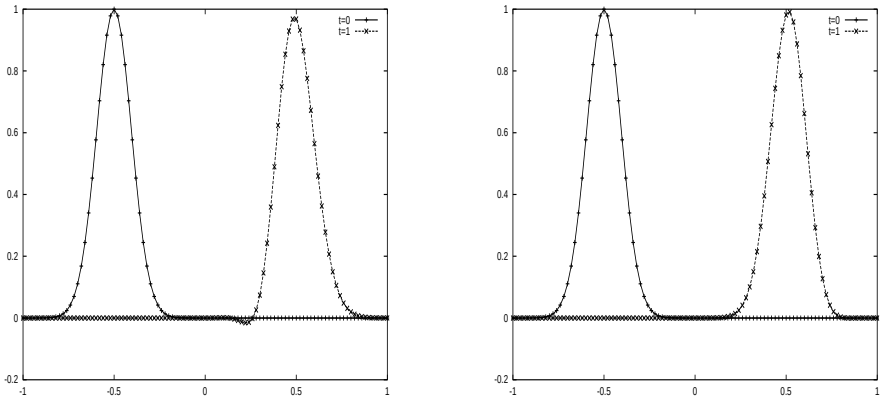
FIGURE 4.8 *Phase errors for Lax-Wendroff time discretization, plot against kh, for* $\lambda = \frac{\Delta t}{h} = 0.25, 0.50, 0.75$

the numerical solution is moving faster than the Fourier mode of the exact solution since the phase errors are negative for $\lambda = 0.25, 0.5, 0.75$.

Next we show the application of IMLS method for linear advection equation. For the purpose of comparison, we take the initial data as in Kunle's work [Kun01].

$$u(x,0) = e^{-55(x+0.5)^2}, \quad -1 \leq x \leq 1. \tag{4.15}$$

The parameter of mesh ratio in this test was chosen as $\lambda = 0.9$, the final time $t = 1$ and $N = 100$. We display the outputs shown in Figure 4.9(a) of LW-IQS scheme and Figure 4.9(b) of LW-IQNS scheme.



(a) LW-IQS scheme (b) LW-IQNS scheme

FIGURE 4.9 *Results at $T = 1$ of linear advection equation with initial data.*

The numerical output from the LW-IQNS scheme does not produce the spurious oscillation. Moreover, for the computation of the spatial derivative u_x, IQNS scheme only contains points in the upwind side whereas the IQS scheme includes points in downwind side (see Figure 4.2). However, there is an alternative to improve the LW-IQS scheme by using higher order basis polynomials for constructing IMLS interpolants. This idea has already been proposed by Kunle [Kun01].

TABLE 4.6
Components of amplification factors of Fourier modes for the Lax-Wendroff scheme

Method	LW-FDM	LW-IQS	LW-IQNS
Real parts	$1-2\lambda^2\sin^2\left(\frac{kh}{2}\right)$	$1+\lambda^2(\cos(kh)-1)$	$1-\frac{\lambda}{2}(3-4\cos(kh)+\cos(2kh))$
		$+\frac{4\lambda^2}{\xi^{a}+16}\left(\cos^2(kh)-4\cos(kh)+2\right)$	$-\frac{3\lambda(\xi+4)}{2(9+9\xi+\xi_1{}^{b})}(1-3\cos(kh)+3\cos(2kh)-\cos(3kh))$
			$+\frac{\lambda^2}{2}(1-2\cos(kh)+\cos(2kh))$
			$-\frac{3\lambda^2(\xi+2)}{2(9+9\xi+\xi_1)}(1-3\cos(kh)+3\cos(2kh)-\cos(3kh))$
Imaginary parts	$-\lambda\sin(kh)$	$-\lambda\sin(kh)$	$-\frac{\lambda}{2}(4\sin(kh)-\sin(2kh))$
		$-\frac{4\lambda}{4+\xi}(\sin(kh)(\cos(kh)-1))$	$+\frac{3\lambda(\xi+4)}{2(9+9\xi+\xi_1)}(-3\sin(kh)+3\sin(2kh)-\sin(3kh))$
			$+\frac{\lambda^2}{2}(2\sin(kh)-\sin(2kh))$
			$-\frac{3\lambda^2(\xi+2)}{2(9+9\xi+\xi_1)}(3\sin(kh)-3\sin(2kh)+\sin(3kh))$

[a] $\xi=\frac{w_{k-1}}{w_{k-2}}$
[b] $\xi_1=\frac{w_{k-1}}{w_{k-3}}$

Chapter 5

High order reconstruction using IMLS derivatives

Reconstruction is a key point in high resolution schemes and usually addressed by substituting the piecewise polynomials of the field variables inside each computational region by higher order polynomials. The basic first-order scheme is based on piecewise constant polynomials. The limitation of the first order schemes is generally the absence of sufficient accuracy. Because of this reason, high order schemes have been introduced and developed by a number of researchers in the literature.

In this chapter, we present our new schemes build on meshfree-IMLS interpolation. For the high order method, a WENO concept will be applied to the method in order to avoid spurious oscillation and to achieve the desired high order accuracy. Our schemes are based on the work of Levy *et al.* [LPR00]. In this study the reconstruction will be seperated into two parts: the pointwise reconstruction and its first derivatives.

The smoothness indicator of WENO schemes definded by Friedrich [Fri98, Fri99] will be introduced in this study. To verify the efficiency of this indicator, the indicator developed by Jiang and Shu [JS96] will also be discussed.

5.1 Second order reconstruction

This section presents the linear reconstructions using Taylor series expansion. The first derivatives of the field variables will be computed using IMLS approximation. Consider a uniform spatial grid where the stencil $I_j = \left[x_{j-\frac{1}{2}}, x_{j+\frac{1}{2}}\right]$ has a width h.

Let $\chi_j(x)$ be the characteristic function of the stencil I_j where

$$\chi_j(x) = \begin{cases} 1 & \text{if } x \in I_j \\ 0 & \text{if } x \notin I_j. \end{cases}$$

A general piecewise polynomial is of the form

$$\tilde{R}(x) = \sum_{i=1}^{N} P_i(x)\chi_i(x). \tag{5.1}$$

The linear reconstruction of the field variable can be written as

$$P_i(x) = u_i + u_i'(x - x_i) \tag{5.2}$$

where u_i is the given point values and u_i' is the first derivative of u at x_i. We want to reconstruct piecewise polynomial (5.1) by applying the IMLS method.

Theorem 5.1. *Assume $f(x)$ be a smooth function over the interval $[x_l, x_r]$. Given an integer N, let $x_l = x_1 < x_2 < \ldots < x_N < x_{N+1} = x_r$ be a partition of the interval $[x_l, x_r]$ with $x_i - x_{i-1} = h$ for $i = 2, \ldots, N+1$. Then the L_2-error of the linear function is*

$$err = \sqrt{\frac{1}{x_{j+1} - x_j} \int_{x_j}^{x_{j+1}} |f(x) - P_j(x)|^2 dx}, \tag{5.3}$$

where

$$P_j(x) = f_j + \frac{d}{dx} Lf(x_j)(x - x_j). \tag{5.4}$$

$\frac{d}{dx} Lf(x)$ is an approximated slope obtained by IMLS approximations. We consider these approximants in three cases based on the different window functions with a fixed dilation parameter $R = 3h$,(see section 2.2 in (2.25)- (2.27)) as follows:

1. *If $\frac{d}{dx} Lf(x_j) = \sum_{i=1}^{N} \frac{d}{dx}\phi_i^l(x_j) f_i$, then the L_2-error is*

$$err = h^2 \sqrt{\frac{(f_i'')^2}{320} + \frac{(f_{i-1}'')^2}{3(\xi + 4)^2}}, \tag{5.5}$$

2. *if $\frac{d}{dx} Lf(x_j) = \sum_{i=1}^{N} \frac{d}{dx}\phi_i^f(x_j) f_i$, then the L_2-error is*

$$err = h^2 \sqrt{\frac{(f_i'')^2}{320} + \frac{h^2 (f_i''')^2}{3(\xi + 4)^2}}, \tag{5.6}$$

and

3. if $\frac{d}{dx}Lf(x_j) = \sum_{i=1}^{N} \frac{d}{dx}\phi_i^r(x_j)f_i$*, then the* L_2*-error is*

$$err = h^2\sqrt{\frac{(f_i'')^2}{320} + \frac{(f_{i+1}'')^2}{3(\xi+4)^2}}. \tag{5.7}$$

Proof. We will only consider the first case, using Taylor expansion, $f(x)$ can be expressed as

$$f(x) = f(x_i) + f'(x_i)(x-x_i) + \frac{1}{2}f''(x_i)(x-x_i)^2 + \frac{1}{6}f'''(\zeta)(x-x_i)^3, \tag{5.8}$$

where ζ is a certain number in the interval (x_j, x_{j+1}). The difference between $f(x)$ and $P_j(x)$ is

$$f(x) - P_j(x) = (f_i' - \sum_{j=1}^{N} \frac{d}{dx}\phi_j^l(x_i)f_j)(x-x_i) + \frac{f_i''}{2}(x-x_i)^2 + \mathcal{O}(h^3). \tag{5.9}$$

From (3.6) and lemma 3.3, we obtain

$$\begin{aligned}\sum_{j=1}^{N} \frac{d}{dx}\phi_j^l(x_i)f_j &= \frac{f_i - f_{i-1}}{h} - \frac{2(f_i - 2f_{i-1} + f_{i-2})}{h(4+\xi)} \\ &= f'(x_i) - \frac{2hf_{i-1}''}{4+\xi}.\end{aligned} \tag{5.10}$$

Substituting the absolute value of the right hand sides of (5.9) into (5.3), we obtain

$$err^2 = \frac{1}{x_{j+1}-x_j}\int_{x_j}^{x_{j+1}} \left(\frac{2hf_{i-1}''}{\xi+4}(x-x_i) + \frac{f_i''}{2}(x-x_i)^2 + \mathcal{O}(h^3)\right)^2 dx \tag{5.11}$$

Let x_i be the midpoint value of the interval $[x_j, x_{j+1}]$, then the explicit integral of (5.11) is

$$err^2 = h^4\frac{(f_{i-1}'')^2}{3(\xi+4)^2} + h^4\frac{(f_i'')^2}{320} + \mathcal{O}(h^5).$$

Therefore,

$$\begin{aligned}err &= \sqrt{h^4\frac{(f_{i-1}'')^2}{3(\xi+4)^2} + h^4\frac{(f_i'')^2}{320} + \mathcal{O}(h^5)} \\ &= h^2\sqrt{\frac{(f_{i-1}'')^2}{3(\xi+4)^2} + \frac{(f_i'')^2}{320} + \mathcal{O}(h)}.\end{aligned}$$

■

The result from Theorem 5.1 points out that linear reconstruction by using the first order IMLS derivatives based on either symmetric or non-symmetric window

function results in second order accuracy.

5.2 Third order reconstruction

In this section, the higher order scheme deals with the first derivatives computed by IMLS method based on quadratic basis and symmetric window function.
Using a Taylor series expansion, the quadratic reconstruction reads

$$P_i(x) = u_i + u_i'(x - x_i) + \frac{1}{2}u_i''(x - x_i) \tag{5.12}$$

where u_i is the given point value, u_i' and u_i'' are the first and the second derivative of u at x_i, respectively. We want to reconstruct piecewise quadratic function by applying the first and the second derivatives computed from IMLS method as follows:

Lemma 5.2. *Let $f(x)$ be a smooth function over the interval $[x_l, x_r]$. Given an integer N, let $x_l = x_1 < x_2 < \ldots < x_N < x_{N+1} = x_r$ be a partition of interval $[x_l, x_r]$ with $x_i - x_{i-1} = h$ for $i = 2, \ldots, N+1$. The L_2-error of the quadratic function is*

$$err = \sqrt{\frac{1}{x_{j+1} - x_j}\int_{x_j}^{x_{j+1}} |f(x) - P_j(x)|^2 dx}, \tag{5.13}$$

where

$$P_j(x) = f_j + \frac{d}{dx}Lf(x_j)(x - x_j) + \frac{1}{2}\frac{d^2}{dx^2}Lf(x_j)(x - x_j)^2. \tag{5.14}$$

$\frac{d}{dx}Lf(x_j)$ and $\frac{d^2}{dx^2}Lf(x_j)$ are approximated derivatives obtained by IMLS approximations based on quadratic basis and symmetric window function with a fixed dilation parameter $R = 3h$. The approximated derivatives may be given as

$$\frac{d}{dx}Lf(x_j) = \sum_{i=1}^{N}\frac{d}{dx}\phi_i^f(x_j)f_i \quad \frac{d^2}{dx^2}Lf(x_j) = \sum_{i=1}^{N}\frac{d^2}{dx^2}\phi_i^f(x_j)f_i. \tag{5.15}$$

Then the L_2-error is

$$err = h^3\sqrt{\frac{(f_i''')^2}{3(4+\xi)^2} + \frac{h^2(f_i^{(4)})^2}{20(16+\xi)^2}}. \tag{5.16}$$

Proof. Using Taylor expansion of $f(x)$, the absolute value of the difference between $f(x)$ and $P_j(x)$, $|f(x) - P_j(x)|$, is

$$\left|(f_i' - \sum_{j=1}^{N}\frac{d}{dx}\phi_j^f(x_i)f_j)(x - x_i) + \frac{1}{2}(f_i'' - \sum_{j=1}^{N}\frac{d^2}{dx^2}\phi_j^f(x_i)f_j))(x - x_i)^2 + \mathcal{O}(h^3)\right|. \tag{5.17}$$

According to equation (3.30) and lemma 3.6, the first derivative in the IMLS method

based on symmetric window function can be written as

$$\begin{aligned}\sum_{j=1}^{N}\frac{d}{dx}\phi_j^f(x_i)f_j &= \frac{f_{i+1}-f_{i-1}}{2h}+\frac{(f_{i+2}-2f_{i+1}+2f_{i-1}-f_{i-2})}{h(4+\xi)}\\ &= f'(x_i)+\frac{2h^2f_i'''}{4+\xi}.\end{aligned} \tag{5.18}$$

Analogously, the second derivative in the IMLS method reads as

$$\begin{aligned}\sum_{j=1}^{N}\frac{d^2}{dx^2}\phi_j^f(x_i)f_j &= \frac{f_{i+1}-2f_i+f_{i-1}}{h^2}+\frac{4(f_{i+2}-4f_{i+1}+6f_i-4f_{i-1}+f_{i-2})}{h^2(16+\xi)}\\ &= f''(x_i)+\frac{4h^2f_i^{(4)}}{16+\xi}.\end{aligned} \tag{5.19}$$

Substituting equation (5.18) and (5.19) into (5.13), the square of L_2 error of the quadratic function $P(x)$ is

$$err^2=\frac{1}{x_{j+1}-x_j}\int_{x_j}^{x_{j+1}}\left(\frac{2h^2f_i'''}{\xi+4}(x-x_i)+\frac{2h^2f_i^{(4)}}{\xi+16}(x-x_i)^2+\mathcal{O}(h^3)\right)^2dx \tag{5.20}$$

Let x_i be the midpoint of the interval $[x_j, x_{j+1}]$, then the computation of the integral in (5.20) gives

$$err^2=h^6\frac{(f_i''')^2}{3(4+\xi)^2}+h^8\frac{(f_i^{(4)})^2}{20(16+\xi)^2}.$$

Therefore,

$$\begin{aligned}err&=\sqrt{h^6\frac{(f_i''')^2}{3(4+\xi)^2}+h^2\frac{(f_i^{(4)})^2}{20(16+\xi)^2}}\\ &=h^3\sqrt{\frac{(f_i''')^2}{3(4+\xi)^2}+h^2\frac{(f_i^{(4)})^2}{20(16+\xi)^2}}.\end{aligned}$$

■

Lemma 5.2 points out that the quadratic polynomial reconstruction with the use of the derivatives computed from IMLS methods results in third order accuracy.

5.3 The CWENO-IMLS (CWI) Reconstruction

In Section 5.1 and Section 5.2, we introduced the second order and the third order reconstructions. Generally, the use of unlimited high order reconstructions may lead to oscillatory solutions in the presence of large gradients. Several strategies

were suggested to introduce the concept of upwinding to prevent the creation and evolution of spurious oscillations. Essentially Non-Oscillatory (ENO) schemes are generally based on a reconstruction procedure. The key idea of the r-th order ENO presented in [HEOC87] is a selection of the r possible stencils. The procedure of selection is based on minimizing the spurious numerical oscillations which can be created due to the approximation of function and/or its derivatives in non-smooth regions.

Later, Liu, Osher and Chan [LOC94] developed the Weighted Essentially Non-Oscillatory (WENO) reconstruction which takes a convex combination of all r candidate stencils. Jiang and Shu [JS96] designed the proper smoothness indicators so that the reconstruction achieves $(2r-1)$-th order spatial accuracy in smooth region and switches to the r-th order ENO reconstruction near discontinuities.

Central WENO (CWENO) schemes have been developed by Levy *et.al* [LPR99a]. The advantages of CWENO schemes are that they are very easy to use and can be easily adapted to a large variety of systems without any knowledge of the solution of the Riemann problem. The Lax-Friedrich (LxF) scheme is a well known first order central scheme. It is the forerunner of central schemes and is based on piecewice constant data. A brief description of LxF scheme will be presented in Chapter 6 section 6.2.

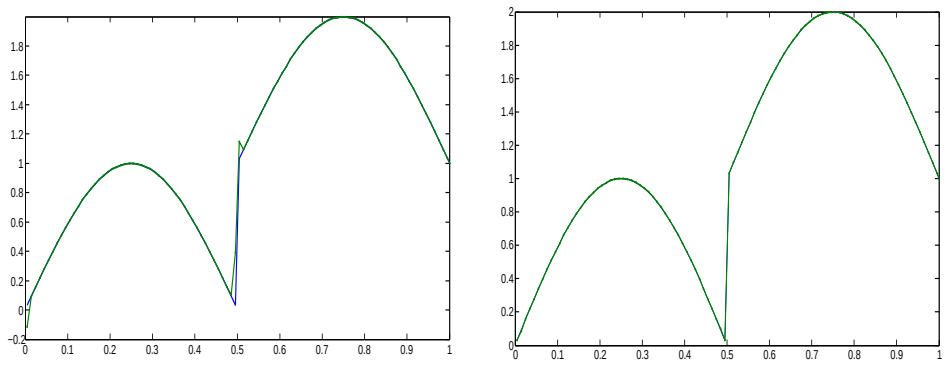
In this section, the concept of CWENO schemes will be combined with our second and third order reconstruction.
To show the properties of the WENO idea, we consider the test problem [JS96]:

$$f(x_j) = \begin{cases} \sin(2\pi x_j) & 0 \leq x_j < 0.5 \\ 1 - \sin(2\pi x_j) & \text{otherwise.} \end{cases} \tag{5.21}$$

Figure 5.1(a) shows the result obtained by using the method descreibed in Section 5.1 whereas Figure 5.1(b) illustrates the result without spurious oscillations near discontinuity $x = 0.5$. These oscillations could be eliminated by the help of the WENO schemes.

5.3.1 Second order Central Weno Interpolating moving least squares (CWI2) reconstruction

We derive our new second order CWI2 reconstruction in one space dimension. We first note that in the absence of large gradients and/or in the smooth region, we get the second order accuracy. However, when large gradients or discontinuities occur, this reconstruction would be oscillatory. Therefore following the WENO



(a) IMLS reconstruction (b) IMLS-WENO reconstruction

FIGURE 5.1 *Comparison between the method with and without WENO concept.*

philosophy [LOC94, JS96], a second order essentially non-oscillatory reconstruction is constructed by taking a convex combination of two linear polynomials which are based on two different stencils. In particular, in the stencil I_j we get

$$\tilde{R}(x) = \omega_l P_l(x) + \omega_r P_r(x), \quad \omega_l + \omega_r = 1, \quad \omega_l, \omega_r \geq 0 \tag{5.22}$$

where $P_l(x)$ and $P_r(x)$ are linear polynomials based on a left stencil and a right stencil, respectively and defined in Theorem 5.1, equation (5.14). Rewrite the two linear functions $P_l(x)$ and $P_r(x)$ as

$$P_l(x) = f_j + \tilde{f}'_l(x - x_j) \quad \text{and} \quad P_r(x) = f_j + \tilde{f}'_r(x - x_j)$$

where

$$\tilde{f}'_l = \sum_{i=1}^{N} \frac{d}{dx}\phi_i^l(x) f_i \quad \tilde{f}'_r = \sum_{i=1}^{N} \frac{d}{dx}\phi_i^r(x) f_i.$$

With an appropriate definition of nonlinear weights ω_j it turns out that the WENO scheme achieves an optimal accuracy in smooth region while preserving essentially non-oscillatory property near discontinuities.

Following the ideas of [JS96], the weights ω_j are defined as

$$\omega_j = \frac{c_j(\epsilon + IS_j)^{-2}}{\sum_k c_k(\epsilon + IS_k)^{-2}}, \tag{5.23}$$

where c_j's are the constant coefficients defined as $c_l = c_r = \frac{1}{2}$ and IS_j are so called smoothness indicators of the stencil I_j. The smoothness indicators IS_j from [JS96]

measure the L_2 norms of all derivatives and are given by

$$IS_i = \sum_{l=1}^{2} \int_{x_{j-1/2}}^{x_{j+1/2}} h^{2l-1} \left(\frac{d^{(l)}}{dx^{(l)}} P_i(x) \right)^2 dx, \quad i \in \{l, r\}. \tag{5.24}$$

There are several different ways to compute these indicators suggested in [LOC94, Fri98]. For instance, the indicators defined by Friedrich [Fri98, Fri99] are determined by using L_2 norm of the first derivatives of the polynomials as

$$OI_i = \int_{x_{j-1/2}}^{x_{j+1/2}} h \left(\frac{d}{dx} P_i(x) \right)^2 dx \quad i \in \{l, r\}. \tag{5.25}$$

The indicators in (5.24) are identical to the indicators in (5.25) for the piecewise linear reconstruction. More details concerning these indicators will be disscussed in Subsection 5.3.2. Here, we take $\epsilon = 10^{-4}$ in order to prevent the denominator to become zero.

An explicit integration of (5.24) yields

$$\begin{aligned} IS_l &= \left(f_j - f_{j-1} + \frac{2(-f_j + 2f_{j-1} - f_{j-2})}{4 + \xi} \right)^2 \\ IS_r &= \left(f_{j+1} - f_j + \frac{2(f_j + 2f_{j+1} - f_{j+2})}{4 + \xi} \right)^2. \end{aligned} \tag{5.26}$$

In smooth regions, Taylor expansion of (5.26) gives

$$\begin{aligned} IS_l &= \left(hf'_j - \frac{2h}{4 + \xi} f''_{j-1} \right)^2 \\ IS_r &= \left(hf'_j + \frac{2h}{4 + \xi} f''_{j+1} \right)^2. \end{aligned} \tag{5.27}$$

Reconstruction of the derivatives of point values

Let the function $f(x)$ be approximated by the CWI2 reconstruction in (5.22). A pointwise reconstruction of the first derivative $f'(x)$ is given by

$$\tilde{R}'(x) = \omega_l P'_l(x) + \omega_r P'_r(x) \tag{5.28}$$

where

$$P'_l(x) = \tilde{f}'_l = \sum_{i=1}^{N} \frac{d}{dx} \phi_i^l(x) f_i, \quad P'_r(x) = \tilde{f}'_r = \sum_{i=1}^{N} \frac{d}{dx} \phi_i^r(x) f_i \tag{5.29}$$

and the weights ω_l and ω_r defined in (5.23).

Lemma 5.3. *Assume the reconstruction of the first derivative $f'(x)$ defined in* (5.28) *is in the smooth region such that the weights ω_l and ω_r are equal to the constant*

coefficients $c_l = \frac{1}{2}$ *and* $c_r = \frac{1}{2}$. *Then the reconstruction* (5.28) *results in the second order accurcy.*

Proof. An explicit expression of $\tilde{f}'_r(x_j)$ is

$$\tilde{f}'_r(x_j) = \frac{f_{j+1} - f_j}{h} + \frac{2(f_j - 2f_{j+1} + f_{j+2})}{h(4+\xi)}.$$

Using Lemma 3.1 in Chapter 3 for an explicit form of $\tilde{f}'_l(x_j)$ and taking a combination of these two formulae, we get in the smooth region

$$\tilde{R}'(x) = \frac{f_{j+1} - f_{j-1}}{2h} + \frac{f_{j+2} - 2f_{j+1} + 2f_{j-1} - f_{j-2}}{h(4+\xi)}.$$

We can conclude that the reconstruction in (5.28) results in the second order accuacy according to Lemma 3.5 in equation (3.30) and Lemma 3.6 in Chapter 3. ■

5.3.2 Third order Central Weno Interpolating moving least squares(CWI3) reconstruction

As the result of Lemma 5.2, the higher order scheme can be constructed by the quadratic function $P_o(x)$ as

$$P_o(x) = f_j + \sum_{i=1}^{N} \frac{d}{dx}\phi_i^f(x_j) f_i(x - x_j) + \frac{1}{2}\sum_{i=1}^{N} \frac{d^2}{dx^2}\phi_i^f(x_j) f_i(x - x_j)^2. \tag{5.30}$$

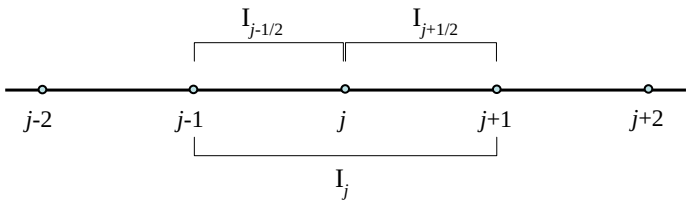
Defining the non-oscillatory reconstruction on the stencil I_j shown in Figure 5.2(a), the following properties should be verified:

$$f(x) = \begin{cases} P_o(x) & \text{if the solution is smooth in interval } I_j, \\ P_l(x) & \text{if the solution is not smooth in the interval } I_{j+1/2}, \\ P_r(x) & \text{if the solution is not smooth in the interval } I_{j-1/2}. \end{cases} \tag{5.31}$$

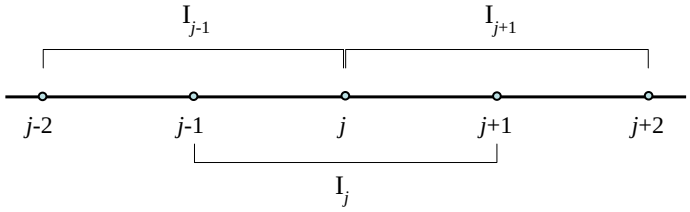
First, it is necessary to find the polynomials $P_l(x)$ and $P_r(x)$. These two linear polynomials $P_l(x)$ and $P_r(x)$ are constructed as in Theorem 5.1. Rewrite both linear polynomials as

$$\begin{aligned} \tilde{f}_l(x) = P_l(x) &= f_j + \sum_{i=1}^{N} \frac{d}{dx}\phi_i^l(x_j) f_i(x - x_j) \\ \tilde{f}_r(x) = P_r(x) &= f_j + \sum_{i=1}^{N} \frac{d}{dx}\phi_i^r(x_j) f_i(x - x_j). \end{aligned} \tag{5.32}$$

In order to avoid the spurious oscillations near discontinuities or in the presence of large gradients, the notion of WENO scheme will be applied. By taking a convex



(a) The second order method which is of the third order only if the stencil I_j is used



(b) The third order method

FIGURE 5.2 *Local domain difference.*

combination of polynomials which depend on different stencils, we get

$$\tilde{R}(x) = \sum_i \omega_i P_i(x) \quad \sum_i \omega_i = 1 \quad \omega_i \geq 0 \quad i \in \{l, c, r\} \tag{5.33}$$

where P_l and P_r are linear polynomials defined in (5.32) corresponding to the $I_{j-1/2}$ stencil and the $I_{j+1/2}$ stencil, respectively. It is our goal to control the weights ω_i such that $\tilde{R}(x) = P_o(x)$ so that we would have a piecewise quadratic polynomial reconstruction in the smooth regions and switch to one of the one-sided linear functions near discontinuities .

In general, the nonlinear weights ω_j in (5.23) must smoothly converge to the ideal constants c_j in (5.36) as h approaches zero in the smooth regions. On the other hand, in regions where discontinuity does exist, the nonlinear weights ω_j of the quadratic polynomial and of one of the one-sided linear reconstruction tend to be zero. Then the second order method becomes the first order method according to properties (5.31).

Assume that the solution is smooth over the stencil I_j, then the nonlinear weights ω_i act as the constant coefficients c_j, and (5.33) yields

$$\tilde{R}(x) = P_o(x) = c_l P_l(x) + c_c P_c(x) + c_r P_r(x), \quad \sum_i c_i = 1 \quad i \in \{l, c, r\}. \tag{5.34}$$

Then we can find the polynomial $P_c(x)$ as:

$$P_c(x) = \frac{1}{c_c}(P_o(x) - c_l P_l(x) - c_r P_r(x)). \tag{5.35}$$

Refering to the paper [LPR00], any symmetric choice of constants c_i in (5.34) results in third order accuracy. The linear constant coefficients c_j's are given by

$$c_l = c_r = \frac{1}{4} \quad \text{and} \quad c_c = \frac{1}{2}. \tag{5.36}$$

Then the central polynomial $P_c(x)$ is

$$\begin{aligned} P_c(x) &= 2P_o(x) - \frac{1}{2}(P_l(x) + P_r(x)) \\ &= f_j + (2\tilde{f}'_f - \frac{1}{2}\tilde{f}'_l - \frac{1}{2}\tilde{f}'_r)(x - x_j) + \tilde{f}''_f(x - x_j)^2. \end{aligned} \tag{5.37}$$

where

$$\tilde{f}'_l = \sum_{i=1}^{N} \frac{d}{dx}\phi_i^l(x) f_i, \qquad \tilde{f}'_r = \sum_{i=1}^{N} \frac{d}{dx}\phi_i^r(x) f_i, \tag{5.38}$$

$$\tilde{f}'_f = \sum_{i=1}^{N} \frac{d}{dx}\phi_i^f(x) f_i, \qquad \tilde{f}''_f = \sum_{i=1}^{N} \frac{d^2}{dx^2}\phi_i^f(x) f_i. \tag{5.39}$$

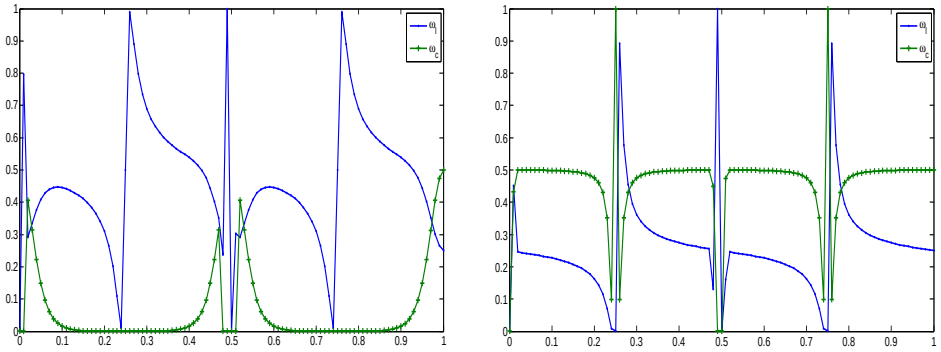
In particular, $\tilde{f}'_l$ and $\tilde{f}'_r$ are computed by linear-IMLS method while $\tilde{f}'_f$ and $\tilde{f}''_f$ are determined by quadratic-IMLS method.

Rewrite the smoothness indicators IS_j as

$$IS_i = \sum_{l=1}^{2} \int_{x_{j-1/2}}^{x_{j+1/2}} h^{2l-1} \left(\frac{d^{(l)}}{dx^{(l)}} P_i(x) \right)^2 dx, \quad i \in \{l, c, r\}. \tag{5.40}$$

An explicit integration of (5.40) yields

$$\begin{aligned} IS_l &= \left(f_j - f_{j-1} + \frac{2(-f_j + 2f_{j-1} - f_{j-2})}{4 + \xi} \right)^2 \\ IS_c &= \left(\frac{f_{j+1} - f_{j-1}}{2} + \frac{-f_{j-2} + 2f_{j-1} - 2f_{j+1} + f_{j+2}}{4 + \xi} \right)^2 \\ &\quad + \frac{1}{3}\left(f_{j+1} - 2f_j + f_{j-1} + \frac{4(f_{j-2} - 4f_{j-1} + 6f_j - 4f_{j+1} + f_{j+2})}{16 + \xi} \right)^2 \\ IS_r &= \left(f_{j+1} - f_j + \frac{2(f_j + 2f_{j+1} - f_{j+2})}{4 + \xi} \right)^2. \end{aligned} \tag{5.41}$$



(a) IS (b) OI

FIGURE 5.3 *Comparison of nonlinear weights.*

In smooth regions, Taylor expansion of (5.41) gives

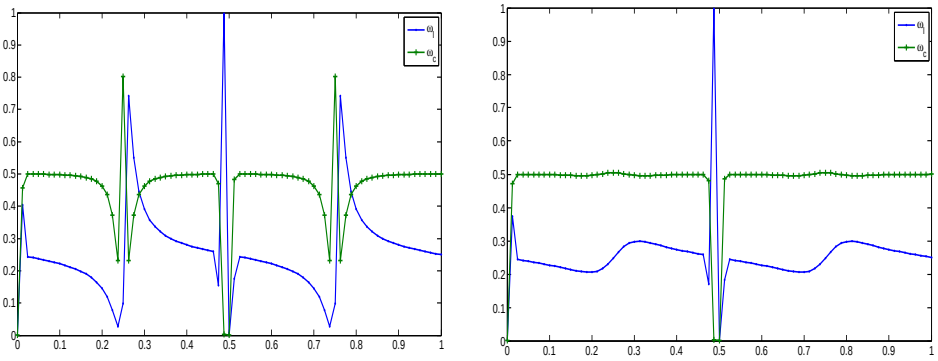
$$\begin{aligned} IS_l &= \left(hf'_j - \frac{2h}{4+\xi} f''_{j-1} \right)^2 \\ IS_c &= \left(hf'_j + \frac{hf'''_j}{(4+\xi)} \right)^2 + \frac{1}{3} \left(h^2 f''_j + \frac{4h^2 f^{(4)}_j}{(16+\xi)} \right)^2 \\ IS_r &= \left(hf'_j + \frac{2h}{4+\xi} f''_{j+1} \right)^2 . \end{aligned} \tag{5.42}$$

To investigate the behaviour of the nonlinear weights (5.23), we display the weights ω_l, ω_c in Figure 5.3(a) which refer to the computation of the function in (5.21) on the interval $[0, 1]$ with $N = 100$. We set $\epsilon = 10^{-6}$ in (5.23). Note that in the smooth region the optimal weights are $\omega_l \approx c_l = \frac{1}{4}$ and for $\omega_c \approx c_c = \frac{1}{2}$. We can see from Figure 5.3(a) that although the solution is in the smooth region, the values of central weights ω_c are practically zero and the values of weights ω_l do not converge to the optimal linear constant c_l. Therefore, the method cannot achieve the desired order of accuracy. In the opposite way, the method based on the new weights OI_j see Figure 5.3 seems to be a better alternative.

The smoothness measurements (OI) defined by Friedrich [Fri98, Fri99] are determined by using L_2 norm of the first derivatives of the polynomials as

$$OI_i = \int_{x_{j-1/2}}^{x_{j+1/2}} h \left(\frac{d}{dx} P_i(x) \right)^2 dx \quad i \in \{l, c, r\}. \tag{5.43}$$

In the paper [Fri98], the author pointed out that the solutions from the smoothness indicator OI_i in (5.43) and from indicator IS_i in (5.40) are very similar. But in



(a) $\epsilon = 10^{-6}$ (b) $\epsilon = 10^{-4}$

FIGURE 5.4 *Nonlinear weights.*

our case, they are very different, see Figure 5.3. It seems to be the case that the indicator IS_i is not suitable for the resolution techniques by using IMLS derivatives. Therefore we use the expression of the indicators of the weights described in [Fri98] is

$$\omega_j = \frac{c_j(\epsilon + OI_j)^{-2}}{\sum_k c_k(\epsilon + OI_k)^{-2}}. \tag{5.44}$$

The direct computation of (5.43) yields

$$\begin{aligned} OI_l &= IS_l \\ OI_c &= \left(\frac{f_{j+1} - f_{j-1}}{2} + \frac{-f_{j-2} + 2f_{j-1} - 2f_{j+1} + f_{j+2}}{4 + \xi}\right)^2 \\ OI_r &= IS_r. \end{aligned} \tag{5.45}$$

In smooth regions, Taylor expansion gives

$$OI_c = \left(hf_j' + \frac{hf_j'''}{(4+\xi)}\right)^2.$$

Next, the choice of the parameter ϵ to be specified in (5.44) will be discussed. We again display the weights ω_j in (5.44) with mesh width $h = \frac{1}{40}$ and we use the function in (5.21) as a test function. Figure 5.4 shows the behaviour of the weights ω_j with two different parameters ϵ. Figure 5.4(a) shows the output from $\epsilon = 10^{-6}$ and Figure 5.4(b) shows the results from $\epsilon = 10^{-4}$. It can be seen that the weights computed with $\epsilon = 10^{-6}$ are optimal on the computed domain except around the critical points, more precisely at $x = \frac{1}{4}, \frac{3}{4}$. Compared with Figure 5.4(b), if we increase the magnitude of ϵ from 10^{-6} to 10^{-4}, the weights ω_c and ω_l are close to the linear constant c_c, c_l, respectively. However, around the discontinuity $x = \frac{1}{2}$ and at

the most left grid point, the weights with 10^{-6} and 10^{-4} act similarly. Therefore, we select $\epsilon = 10^{-4}$ for all computations in this study.

We test the accuracy with the Franke function as defined in (4.1) on the interval $[-1, 1]$. By the numerical experiments, see Table 5.1, it turns out that our proposed method with the indicators OI_j in (5.43) is in fact third order accurate while the other scheme with the indicators IS_j in (5.40) seems to be second order accurate as grid size decreases.

TABLE 5.1
$||f(x) - P_o(x)||_1$ error and order of accuracy ($\epsilon = 10^{-6}$)

N	L_1 error(IS)	L_1 order(IS)	L_1 error(OI)	L_1 order(OI)
20	2.47E-002	-	2.03E-002	-
40	5.50E-003	2.17	4.10E-003	2.31
80	1.10E-003	2.32	7.32E-004	2.49
160	2.21E-004	2.32	1.03E-004	2.82
320	4.62E-005	2.23	1.07E-005	3.23

Reconstruction of the derivatives of point values

This computation is requested for estimating the solutions at the intermediate state ν of a Rung-Kutta time integration scheme discussed in Chapter 6.

Assume the function $f(x)$ be approximated by the CWENO-IMLS reconstruction (5.33), The reconstruction of function $f(x)$ is rewritten as

$$f(x) \approx \tilde{R}(x) = \omega_l P_l(x) + \omega_c P_c(x) + \omega_r P_r(x)$$

where

$$\begin{aligned} P_l(x) &= f_j + \tilde{f}'_l(x - x_j) \\ P_c(x) &= f_j + (2\tilde{f}'_f - \frac{1}{2}\tilde{f}'_l - \frac{1}{2}\tilde{f}'_r)(x - x_j) + \tilde{f}''_f(x - x_j)^2 \\ P_r(x) &= f_j + \tilde{f}'_r(x - x_j) \end{aligned}$$

and the nonlinear weights ω_j defined are in (5.44). An approximation of the first derivative of function $f(x)$ can be determined as follows

$$\tilde{R}'(x) = \omega_l P'_l(x) + \omega_c P'_c(x) + \omega_r P'_r(x) \tag{5.46}$$

where

$$P'_l(x) = \tilde{f}'_l = \sum_{i=1}^{N} \frac{d}{dx}\phi_i^l(x) f_i \tag{5.47}$$

$$P_r'(x) = \tilde{f}_r' = \sum_{i=1}^{N} \frac{d}{dx}\phi_i^r(x) f_i \tag{5.48}$$

and

$$P_c'(x) = 2\tilde{f}_f' - \frac{1}{2}\tilde{f}_l' - \frac{1}{2}\tilde{f}_r' + 2\tilde{f}_f''(x - x_j). \tag{5.49}$$

The computation of the weights in (5.46) is the same as in (5.44) with the smoothness indicators in (5.43). And the constants c_j's are the same as in (5.36).

It is obvious to see that for the pointwise reconstruction, if the solution is smooth in the stencil I_j, then $\tilde{R}'(x)$ produces an approximation of the first derivative of the second order accuracy. On the other hand, if the solution is not smooth either in the stencil $I_{j-1/2}$ or in the stencil $I_{j+1/2}$, then $\tilde{R}'(x)$ results an approximation of the first derivative of the first order accuracy.

5.4 Accuracy Tests

For the numerical accuracy tests, we use again the Franke's function defined in (4.1).

Table 5.2 and Table 5.3 show the numerical errors and numerical order of accuracy from linear reconstructions with non-symmetric IMLS and symmetric IMLS method. Both cases produce the second order numerical results although the linear reconstruction based on symmetric window function (see Table 5.3) produce the smaller magnitude of errors than non-symmetric window function (see Table 5.2). From Theorem 5.1 we can see that the magnitude of error in equation (5.6) is smaller than in equation (5.5) when $h < 1$ and $h \to 0$.

TABLE 5.2
$\|f(x) - P_l(x)\|$ and order of accuracy

N	L_1 error	L_1 order	L_2 error	L_2 order	L_∞ error	L_∞ order
20	3.35E-002	-	4.90E-002	-	1.14E-001	-
40	8.60E-003	1.96	1.30E-002	1.91	3.17E-002	1.85
80	2.20E-003	1.97	3.30E-003	1.98	8.40E-003	1.92
160	5.40E-004	2.03	8.24E-004	2.00	2.10E-003	2.00
320	1.35E-004	1.98	2.06E-004	1.98	5.37E-004	1.94

TABLE 5.3
$\|f(x) - P_f(x)\|$ and order of accuracy

N	L_1 error	L_1 order	L_2 error	L_2 order	L_∞ error	L_∞ order
20	1.17E-002	-	1.89E-002	-	4.77E-002	-
40	2.90E-003	2.01	4.50E-003	2.07	1.10E-002	2.12
80	7.10E-003	2.03	1.10E-003	2.03	2.80E-003	1.97
160	1.77E-004	2.01	2.70E-004	2.03	7.04E-004	1.99
320	4.40E-005	1.98	6.73E-005	1.98	1.76E-004	1.98

Table 5.4 and Table 5.5 present the errors and order of accuracy of the CWI2 schemes and the CWI3 schemes, respectively. The numerical results confirm the expected order of accuracy.

TABLE 5.4
The errors and order of accuracy from CWI2 schemes ($\epsilon = 10^{-4}$)

N	L_1 error	L_1 order	L_2 error	L_2 order	L_∞ error	L_∞ order
20	2.47E-002	-	4.06E-002	-	1.25E-001	-
40	5.10E-003	2.28	9.10E-003	2.16	3.39E-002	1.89
80	8.67E-004	2.56	1.71E-003	2.42	6.50E-003	2.38
160	1.93E-004	2.16	3.15E-004	2.43	1.00E-003	2.70
320	4.90E-002	1.96	7.87E-005	1.98	2.31E-004	2.09

TABLE 5.5
The errors and order of accuracy from CWI3 schemes ($\epsilon = 10^{-4}$)

N	L_1 error	L_1 order	L_2 error	L_2 order	L_∞ error	L_∞ order
20	1.99E-002	-	3.33E-002	-	1.03E-001	-
40	3.50E-003	2.51	6.40E-003	2.38	2.65E-002	1.96
80	3.92E-004	3.16	7.38E-004	3.12	2.60E-003	3.35
160	3.40E-005	3.53	6.37E-005	3.53	2.63E-004	3.30
320	2.92E-006	3.50	5.22E-006	3.57	2.03E-005	3.66

TABLE 5.6
$\left\| f'(x) - \tilde{R}'(x) \right\|$ *and order of accuracy (CWI2)*

N	L_1 error	L_1 order	L_2 error	L_2 order	L_∞ error	L_∞ order
20	4.96E-001	-	7.65E-001	-	2.10E+000	-
40	1.94E-001	1.36	3.39E-001	1.18	1.22E+000	0.79
80	5.43E-002	1.83	1.09E-001	1.64	4.63E-001	1.39
160	9.27E-003	2.55	1.87E-002	2.55	8.56E-002	2.44
320	1.41E-003	2.69	2.63E-003	2.79	1.12E-002	2.90

TABLE 5.7
$\left\| f'(x) - \tilde{R}'(x) \right\|$ *and order of accuracy (CWI3)*

N	L_1 error	L_1 order	L_2 error	L_2 order	L_∞ error	L_∞ order
20	3.94E-001	-	6.40E-001	-	1.83E+000	-
40	1.43E-001	1.46	2.70E-001	1.24	1.07E+000	0.78
80	3.50E-002	2.03	6.60E-002	2.03	2.80E-001	1.93
160	6.20E-003	2.50	1.14E-002	2.53	4.72E-002	2.57
320	1.10E-003	2.47	2.00E-003	2.48	7.60E-003	2.60

The L_1, L_2, and L_∞ errors and their accuracy of an approximation of the first derivative by CWI2 and CWI3 method are given in Table 5.6 and Table 5.7. The computation is evaluated on the interval $[0, 1]$. It can be seen that both CWI2 and CWI3 schemes produce the second order accuracy for the numerical approximations of the first derivatives.

Chapter 6

Conservative high-order IMLS reconstruction

This chapter will be concerned with the new second order and third order reconstruction applied on hyperbolic conservation problem. Generally, the methods for approximating solutions of such problems can be divided into two main categories: upwind and central schemes. The first-order Godunov method is the prototype of upwind schemes. There are a variety of high-resolution Godunov-type methods which are based on the reconstruction of a piecewise polynomial approximation from cell averages and evolved exactly to the next time level.

Alternatively, central schemes offer the principal advantage due to their simplicity because no need of specific knowledge of the eigenstructure of a given problem, no approximation Riemann solvers are involved in their construction. As an extension of the staggered version of Lax-Friedrich (LxF), the second order central schemes have been introduced by Nessyahu and Tadmor [NT90] and the sophisticated version by Kurganov and Tadmor [KT00]. A brief review of upwind and central schemes will be presented in Section 6.1.

Higher order schemes in the central framework have been introduced by Bianco, Puppo, and Russo [BPR99] for the essentially non-oscillatory (ENO) reconstructions and by Levy *et al.* [LPR99a, LPR99b] for the WENO approach. A high order central weighted essentially non-oscillatory (CWENO) scheme in [LPR99b] is determined by taking a convex combination of a piecewise parabola function and two one-sided linear functions. An overview of staggered CWENO schemes can be found in [LRR99]. An alternative approach of the staggered high order reconstruction techniques is excluded from the scope of this study. More details of another approach and a collection of the central high order schemes are available at the website for a package of high-resolution central schemes for nonlinear conservation laws and related problems (CentPack) in [CSC].

6.1 Upwind and central schemes

Consider the hyperbolic conservation law

$$u_t + f(u)_x = 0. \tag{6.1}$$

Let Δt be the time step and $\lambda = \Delta t/h$. Assume that grid is uniform, $h = x_{j+1} - x_j$ and $I_j = [x_j - \frac{h}{2}, x_j + \frac{h}{2}]$, then an approximate value of u at time t^n is

$$u_j^n = \frac{1}{h}\int_{x_{j-1/2}}^{x_{j+1/2}} u(x, t^n)dx. \tag{6.2}$$

Integrate (6.1) over I_j and dividing by h, we obtain

$$\frac{d}{dt}u_j = -\frac{1}{h}(f(u(x_{j+\frac{1}{2}}, t)) - f(u(x_{j-\frac{1}{2}}, t)). \tag{6.3}$$

The above equation is a prototype of a large variety of schemes. The flux computation is integrated on the interval $[x_{j-1/2}, x_{j+1/2}]$ where the reconstruction of flux is generally discontinuous (see Figure 6.1(a)). Therefore, the information of wave propagation of Riemann fans is necessarily required to integrate flux along the cell interface in time.

On the other hand, central schemes based on staggered grids require to construct a smooth piecewise polynomial interpolant in the interval I_j. The flux function is integrated over the interval $I_{j+1/2} = [x_j, x_{j+1}]$ where the interpolants are smooth (see Figure 6.1(b)).

The evolution of the point values of the solution u in (6.1) on the staggered grid points reads as

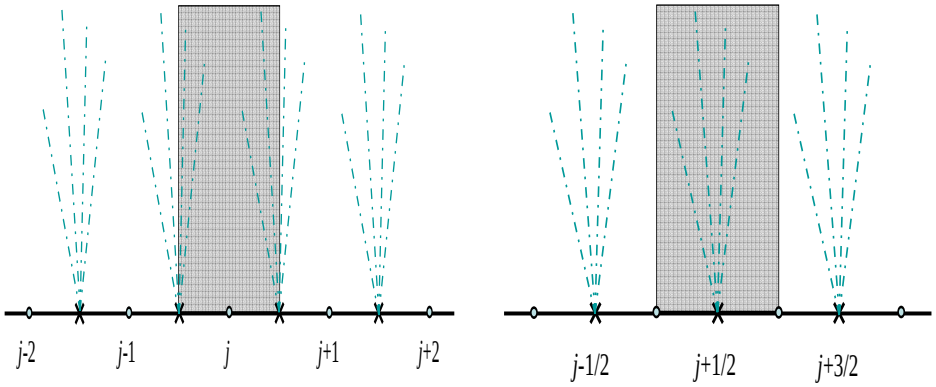
$$\frac{d}{dt}u_{j+1/2} = -\frac{1}{h}(f(u(x_{j+1}, t)) - f(u(x_j, t)). \tag{6.4}$$

The key point of all central schemes is that no requirement of Riemann solvers is needed to evaluate flux function since the Riemann fans are inside computed region of the associated staggered grid. Because of this reason, it makes the computation of the numerical fluxes easier than upwind schemes. More precisely, the computation of flux functions will not require flux splitting.

Integrating (6.1) in space and time and dividing by h, we get

$$u_{j+\frac{1}{2}}^{n+1} = u_{j+\frac{1}{2}}^n - \frac{1}{h}\int_{t^n}^{t^{n+1}} [f(u(x_{j+1}, \tau) - f(u(x_j, \tau))]\, d\tau. \tag{6.5}$$

The first term of the right hand side $u_{j+\frac{1}{2}}^n$ is obtained by integrating the reconstruc-



(a) Upwind scheme (b) Central scheme

FIGURE 6.1 *Flux evaluation between upwind and central schemes.*

tion exactly on $[x_j, x_{j+1}]$ that is

$$u^n_{j+\frac{1}{2}} = \frac{1}{h}\int_{x_j}^{x_{j+1}} u(x,t^n)dx = \frac{1}{h}\left(\int_{x_j}^{x_{j+\frac{1}{2}}} \tilde{R}_j(x)dx + \int_{x_{j+\frac{1}{2}}}^{x_{j+1}} \tilde{R}_{j+1}(x)dx\right). \tag{6.6}$$

A suitable CFL condition for the central scheme is of the form

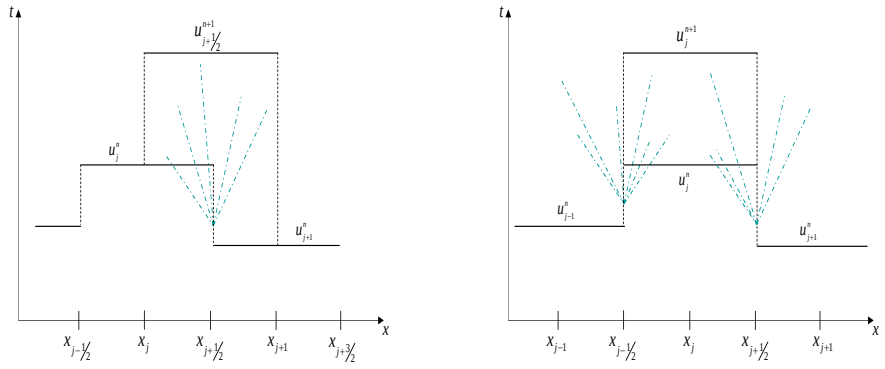
$$\lambda \max_{x_j<x<x_{j+1}} |f'(u(x,t))| \leq \frac{1}{2}. \tag{6.7}$$

6.2 The first order Lax-Friedrich (LxF) scheme

The simplest first order central scheme is Lax-Friedrich (LxF) scheme. This scheme employs piecewise-constant representation of the function on the cell $I_j = [x_{j-\frac{1}{2}} - x_{j+\frac{1}{2}}]$. The resulting LxF scheme is then

$$u^{n+1}_{j+\frac{1}{2}} = \frac{1}{2}(u^n_j + u^n_{j+1}) - \lambda(f(u^n_{j+1} - f(u^n_j)),\ u^n_j := u(x_j,t^n), \tag{6.8}$$

where $\lambda = \Delta t/h$ denotes the mesh ratio and satisfies the CFL condition (6.7). The schematic representation of LxF-staggered version is given in Figure 6.2(a) and non-staggered version in Figure 6.2(b).



(a) Staggered Lax-Friedrich (b) Non-staggered Lax-Friedrich

FIGURE 6.2 *First order Lax-Friedrich schemes.*

6.3 High order central schemes for conservation laws

6.3.1 The method based on NT scheme

Nessyahu-Tadmor (NT) scheme

First, the reconstruction of the field variable will be discussed. Instead of using piecewise constant functions as in LxF scheme, Nessyahu and Tadmor applied a piecewise linear polynomial of the form

$$P_j^n(x) = u_j^n + (u_j')^n(x - x_j) \quad x_{j-1/2} \le x \le x_{j+1/2} \tag{6.9}$$

where $(u_j')^n$ is an approximation of the exact first derivative $u'(x_j, t^n)$. A possible numerical computation of the slope $(u_j')^n$, which prevent spurious oscillations in the numerical solution, is proposed by the concept of limiters. According to the original work of Nessyahu and Tadmor [NT90], the simplest minmod limiter is employed. It is defined as

$$u_j' = \frac{1}{h} MM(u_{j+1} - u_j, u_j - u_{j-1}) \tag{6.10}$$

where

$$MM(a, b) := \frac{1}{2}(\mathsf{sgn}(a) + \mathsf{sgn}(b)) \min(|a|, |b|). \tag{6.11}$$

The staggered point value which is the first integral of the right side of (6.5) is computed by

$$\begin{aligned} u_{j+1/2}^n &= \frac{1}{h} \int_{x_j}^{x_{j+1}} u(x, t^n) dx \\ &= \frac{1}{h} \left(\int_{x_j}^{x_{j+1/2}} P_j(x, t^n) dx + \int_{x_{j+1/2}}^{x_{j+1}} P_{j+1}(x, t^n) dx \right). \end{aligned} \tag{6.12}$$

An explicit integration of (6.12) gives

$$u^n_{j+1/2} = \frac{1}{2}(u^n_j + u^n_{j+1}) - \frac{h}{8}((u'_{j+1})^n - (u'_j)^n) \tag{6.13}$$

where u'_j is defined in (6.10). Next, the evolution step which is the second term of the right side of (6.5) will be described. With the use of a second order quadrature rule such as the midpoint rule and Taylor expansion, the intermediate value is

$$u^{n+1/2}_j = u^n_j - \frac{\lambda}{2}(f'_j)^n \tag{6.14}$$

where f'_j stands for an approximate numerical derivative of the flux and is computed by (6.10). Note that λ satisfies CFL condition (6.7). The final calculation is

$$u^{n+1}_{j+1/2} = u^n_{j+1/2} - \lambda(f(u^{n+1/2}_{j+1}) - f(u^{n+1/2}_j)) \tag{6.15}$$

A schematic representation of NT scheme is shown in Figure 6.3.

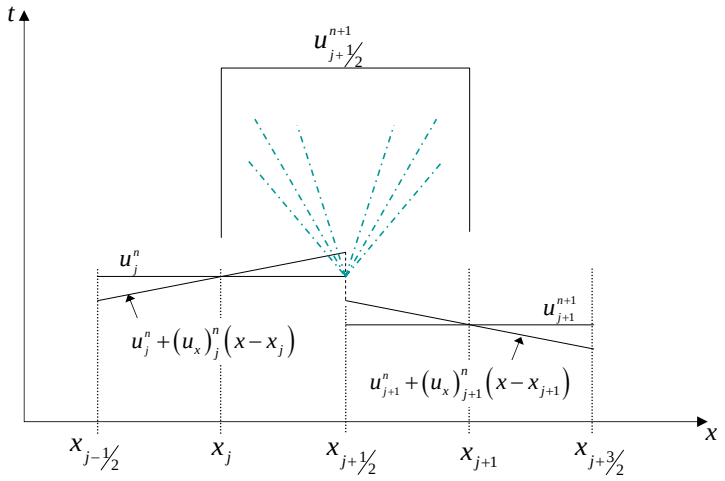


FIGURE 6.3 *NT Schemes.*

NTI scheme

Instead of using the finite difference in (6.10), we apply the derivatives of IMLS interpolations based on linear basis and non-symmetric window functions as represented in Chapter 2. Then the new minmod limiter is

$$u'_j = MM((\tilde{u}'_l)_j, (\tilde{u}'_r)_j) \tag{6.16}$$

where

$$\tilde{u}'_l(x) = \sum_{i=1}^{N} \frac{d\phi_i^l(x)}{dx} u_i \qquad\qquad \tilde{u}'_r(x) = \sum_{i=1}^{N} \frac{d\phi_i^r(x)}{dx} u_i$$

and $MM(a, b)$ is defined in (6.11).

To complete the method, we follow the above description of NT scheme from (6.13) to (6.15). The procedure of NT and NTI schemes can be summerized as a three step method as follow:

1. **Reconstruction step:** Given u_j^n, compute

$$u_{j+1/2}^n = \frac{1}{2}(u_j^n + u_{j+1}^n) - \frac{h}{8}((u'_{j+1})^n - (u'_j)^n)$$

   where u'_j is defined in (6.10) for NT scheme and in (6.16) for NTI scheme.

2. **Predictor step:** Evaluate the intermediate states $u_j^{n+1/2}$

$$u_j^{n+1/2} = u_j^n - \frac{\lambda}{2} f'_j \tag{6.17}$$

   where f'_j is defined in (6.10) for NT scheme and in (6.16) for NTI scheme.

3. **Corrector step:** Update the cell value on the staggered grid

$$u_{j+1/2}^{n+1} = u_{j+1/2}^n - \lambda(f(u_{j+1}^{n+1/2}) - f(u_j^{n+1/2})). \tag{6.18}$$

Remark: To confirm the convergence of the computed solution to the unique entropy solution, an appropriate CFL condition is given by $\lambda \leq 0.32$, For more information and the details of proof, see [NT90].

6.3.2 The Central Runge-Kutta (CRK) schemes

We use the CRK schemes for the second order and the third order time integration. The following discription of CRK schemes is based on [PR06, PPR05, LRR99].

We require that our reconstructions satisfy the following properties:

1. **Accuracy**

   - *Cell averages*

$$u_{j+1/2}^n = \frac{1}{h} \int_{x_j}^{x_{j+1}} u(x, t^n) dx + \mathcal{O}(h^r), \tag{6.19}$$

where r is the spatial order of the method.

- *Pointwise values*

$$P_u(x_j, t^n) = u(x_j, t^n) + \mathcal{O}(h^r). \tag{6.20}$$

- *Derivatives of the numerical fluxes*

$$Df = P'_f(x_j, t^n) = f'(u(x_j, t^n)) + \mathcal{O}(h^{r-1}). \tag{6.21}$$

2. **Conservation**
   For the reconstruction of the function $\tilde{R}$, we want

$$\frac{1}{h}\int_{I_j} \tilde{R}_j(x)dx = u_j \tag{6.22}$$

3. **Non-oscillatory property**
   Prevent the spurious oscillations by means of WENO schemes.

Before going into the detail of CRK schemes, a short review of Runge-Kutta schemes applied to initial value problems is given by discussing the following ordinary differential equation:

$$\begin{aligned} y' &= g(y), \\ y(t_0) &= y_0. \end{aligned}$$

An explicit Runge-Kutta scheme with ν stages for the solution of the initial value problem above reads as

$$y^{n+1} = y^n + \Delta t \sum_{i=1}^{\nu} b_i K^{(i)}$$

where $K^{(i)}$ are so called Runge-Kutta fluxes defined by

$$K^{(i)} = g(y^{(i-1)}) \quad \text{with } y^{(0)} = y^n,\ i = 1, \ldots, \nu.$$

The $y^{(i)}$ are called the intermediate values. In particular, for an explicit schemes these values are given by

$$y^{(i)} = y^n + \Delta t \sum_{l=1}^{i} a_{il} K^{(l)}, \quad i = 1, \ldots, \nu - 1.$$

The matrix $A = (a_{il})$ and the vector $\mathbf{b}$ define the RK scheme. Here the matrix A is a $(\nu - 1) \times (\nu - 1)$ lower triangular matrix with non zero elements in the diagonal.

We now turn out to describe the CRK schemes.

Rewrite the semidiscrete staggered central schemes as

$$\frac{d}{dt}u_{j+1/2}(t) = -\frac{1}{h}[f(u(x_{j+1/2},t) - f(u(x_{j-1/2},t)], \tag{6.23}$$

For time discretization, we apply a ν stage Runge-Kutta scheme. Thus the updated solution can be written as

$$u^{n+1}_{j+1/2} = u^n_{j+1/2} - \lambda \sum_{i=1}^{\nu} b_i K^{(i)}_{j+1/2}, \tag{6.24}$$

and the Rung-Kutta fluxes are

$$K^{(i)}_{j+1/2} = f(u^{(i-1)}_{j+1}) - f(u^{(i-1)}_j) \quad \text{with } u^{(0)}_j = u^n(x_j). \tag{6.25}$$

Now we want to calculate the intermediate values $u^{(i)}_j$ at grid points x_j. Because the reconstruction polynomials are smooth at grid points x_j, we can compute the intermediate values $u^{(i)}_j$ integrating the conservation law (6.1). Then the intermediate states read as

$$u^{(i)}_j = u^{(0)}_j + \Delta t \sum_{l=1}^{i} a_{i,l}\hat{K}^{(l)}_j, \quad i = 1,\dots,\nu-1, \tag{6.26}$$

where

$$\hat{K}^{(l)}_j = -\frac{\partial f(u^{(l-1)})}{\partial x_j} \quad \text{and} \quad u^{(0)}_j = u^n(x_j) \quad l = 1,\dots,\nu. \tag{6.27}$$

The values of the matrix $A = (a_{i,l})$ in (6.26) and the vector $\mathbf{b} = b_i$ in (6.24) are given in Table 6.1.

TABLE 6.1
Coefficients of Rung-Kutta scheme

Scheme	A	$\mathbf{b}$
CRK2	1	$\frac{1}{2}, \frac{1}{2}$
CRK3	1 $\frac{1}{4}, \frac{1}{4}$	$\frac{1}{6}, \frac{1}{6}, \frac{2}{3}$

To finish the computation, we now need to specify the computation of $u^n_{j+1/2}$, point values $u^n(x_j)$, and flux derivatives $f_x(u)$. In all three cases of the computations, the reconstructions are computed as described in Chapter 5, section 5.3. We use the CWI2 descriebed in subsection 5.3.1 for the second order reconstruction and use the CWI3 presented in subsection 5.3.2 for the third order reconstruction.

We start to compute the point values $u^n(x_j)$ and the flux derivatives. Then we

will describe how $u^n_{j+1/2}$ is computed. We rewrite

$$\tilde{R}(x) = \omega_l P_l(x) + \omega_r P_r(x)$$

where

$$P_l(x) = u_j + \tilde{u}'_l(x - x_j)$$
$$P_r(x) = u_j + \tilde{u}'_r(x - x_j)$$

for evaluation of $u^n_{j+1/2}$ and the point value $u^n(x_j)$ of the piecewise linear reconstruction and

$$\tilde{R}(x) = \omega_l P_l(x) + \omega_C P_c(x) + \omega_r P_r(x) \tag{6.28}$$

where

$$P_c(x) = u_j + (2\tilde{u}'_f - \frac{1}{2}\tilde{u}'_l - \frac{1}{2}\tilde{u}'_r)(x - x_j) + \tilde{u}''_f(x - x_j)^2,$$

for the piecewise quadratic reconstruction.

The computation of discrete flux derivatives $\frac{\partial f}{\partial x}$ at the intermediate states in (6.27) is given by

$$\tilde{R}'(x) = \omega_l P'_l(x) + \omega_r P'_r(x) \quad \text{for the CWI2 method and}$$
$$\tilde{R}'(x) = \omega_l P'_l(x) + \omega_c P'_c(x) + \omega_r P'_r(x) \quad \text{for the CWI3 method}$$

where

$$P'_l(x) = \tilde{f}'_l = \sum_{i=1}^{N} \frac{d}{dx}\phi^l_i(x) f_i$$

$$P'_r(x) = \tilde{f}'_r = \sum_{i=1}^{N} \frac{d}{dx}\phi^r_i(x) f_i$$

and

$$P'_c(x) = 2\tilde{f}'_f - \frac{1}{2}\tilde{f}'_l - \frac{1}{2}\tilde{f}'_r + 2\tilde{f}''_f(x - x_j).$$

Note that the reconstruction $\tilde{R}(x)$ defined in (6.28) with the central quadratic function $P_c(x) = u_j + (2\tilde{u}'_f - \frac{1}{2}\tilde{u}'_l - \frac{1}{2}\tilde{u}'_r)(x - x_j) + \tilde{u}''_f(x - x_j)^2$ is not conservative, i.e. $\int_{I_j} \tilde{R}(x) \neq u_j$. The following Lemma states how to compute the conservative version of the reconstruction.

Lemma 6.1. *If the quadratic polynomial $P_c(x)$ defined on the interval $[x_{j-\frac{1}{2}}, x_{j+\frac{1}{2}}]$ is given by*

$$P_c(x) = u_j - \frac{h^2}{12}(\tilde{u}''_f) + (2\tilde{u}'_f - \frac{1}{2}\tilde{u}'_l - \frac{1}{2}\tilde{u}'_r)(x - x_j) + \tilde{u}''_f(x - x_j)^2 \tag{6.29}$$

where $\tilde{u}'_l, \tilde{u}'_r, \tilde{u}'_f, \tilde{u}''_f$ *are defined in* (5.39)*, then*

$$\tilde{R}(x) = \omega_l P_l(x) + \omega_c P_c(x) + \omega_r P_r(x) \tag{6.30}$$

is conservative.

Proof. In the smooth region, the nonlinear weights ω_j act as the constants c_j, see Chapter 5, section 5.3, $\tilde{R}(x)$ is rewritten as

$$\begin{aligned}\tilde{R}(x) &= \frac{1}{4}P_l(x) + \frac{1}{2}P_c(x) + \frac{1}{4}P_r(x) \\ &= u_j - \frac{h^2}{24}\tilde{u}''_f + \tilde{u}'_f(x - x_j) + \frac{\tilde{u}''_f}{2}(x - x_j)^2.\end{aligned}$$

Integrating $\tilde{R}(x)$ on the interval $[x_{j-\frac{1}{2}}, x_{j+\frac{1}{2}}]$ and dividing by h, we obtain

$$\frac{1}{h}\int_{x_{j-1/2}}^{x_{j+1/2}} \tilde{R}(x)dx = \frac{1}{h}\left[(u_j - \frac{h^2 u''_f}{24})x + \frac{\tilde{u}'_f}{2}(x - x_j)^2 + \frac{\tilde{u}''_f}{6}(x - x_j)^3\right]_{x_{j-1/2}}^{x_{j+1/2}}.$$

Substituting $x = x_{j+1/2}$ and $x = x_{j-1/2}$, we find

$$\frac{1}{h}\int_{x_{j-1/2}}^{x_{j+1/2}} \tilde{R}(x)dx = u_j.$$

Assume the solution is not smooth in the interval $I_{j+1/2} = [x_j, x_{j+1}]$, the third order reconstruction $\tilde{R}(x)$ becomes the second order reconstruction corresponding to the interval $I_{j-1/2} = [x_{j-1}, x_j]$. It is obviously seen that the result of the integration of the linear polynomial divided by h is

$$\frac{1}{h}\int_{I_j} \tilde{R}(x)dx = u_j.$$

Therefore, the scheme (6.30) is conservative. ■

Note that it is very obvious to see that the piecewise linear reconstruction is conservative.
We then can compute the $u^n_{j+1/2}$ for the CWI3 reconstruction by using the results of Lemma 6.1 and substituing into the integration in (6.6).

Next, the steps of the CWI Runge-Kutta (CWIRK) schemes can be summarized in the following algorithm:

Given the point value u^n_j at the time step t^n, proceed as follows

- **Step 1:**

$\forall j$: Compute the reconstruction of $u^n_{j+1/2}$, and u^n_j

$$\tilde{R}(x) = \omega_l P_l(x) + \omega_r P_r(x) \quad \text{for the second order reconstruction}$$
$$\tilde{R}(x) = \omega_l P_l(x) + \omega_c P_c(x) + \omega_r P_r(x) \quad \text{for the third order reconstruction.}$$

The polynomials $P_l(x)$ and $P_r(x)$, and $P_c(x)$ are given in (5.32) while the quadratic polynomials $P_c(x)$ are defined in Lemma 6.1 equation (6.29) . The weights ω_j are computed by (5.44) and are depending on the smoothness indicators OI_j in (5.43).

- **Step 2:**
  Compute the point values of the flux $f(u^n_j)$, where the point values u_j are obtained from step 1. Reconstruct the discrete first derivative of $f'(u^n_j)$ by using the point values of $f(u^n_j)$ and follow the reconstruction procedure described in Chapter 5 Section 5.3

  $$f'(x) = \omega_l f'_l(x) + \omega_r f'_r(x) \quad \text{for the second order reconstruction}$$
  $$f'(x) = \omega_l f'_l(x) + \omega_c f'_c(x) + \omega_r f'_r(x) \quad \text{for the third order reconstruction}$$

  where f'_l, f'_r, and f'_c are defined in (5.47)- (5.49). The weights ω_j are computed as in step 1 but the calculation of the smoothness indicators are replaced by the discrete flux f.

- **Step 3:**
  Compute the intermediate values $u^{(i)}_j$. For $i = 1, \ldots, \nu - 1$
  - Evaluate $f(u^{(i-1)}_j)$
  - Compute

    $$u^{(i)}_j = u^{(0)}_j + \Delta t \sum_{l=1}^{i} a_{i,l} \hat{K}^{(l)}_j, \quad i = 1, \ldots, \nu - 1,$$

    where $\hat{K}^{(l)}_j = -\frac{\partial f(u^{(l-1)})}{\partial x_j}$ are obtained by step 2. The coefficents a_{il} are listed in Table 6.1.

- **Step 4:**
  Evaluate the Runge-Kutta fluxes

  $$K^{(i)}_{j+1/2} = f(u^{(i-1)}_{j+1}) - f(u^{(i-1)}_j) \quad i = 1, \ldots, \nu \quad \text{with } u^{(0)}_j = u^n(x_j).$$

- **Step 5:**

Update the staggered $u_{j+1/2}^{n+1}$ at the new time step $n+1$ by

$$u_{j+1/2}^{n+1} = u_{j+1/2}^{n} - \lambda \sum_{i=1}^{\nu} b_i K_{j+1/2}^{(i)},$$

where the coefficients b_i are shown in Table 6.1.

6.3.3 CFL conditions

Rewrite the CFL conditions for all central staggered schemes as

$$\lambda \leq \lambda_0 \frac{1}{\max(|f'(u(x,t))|)} \leq \frac{1}{2} \frac{1}{\max(|f'(u(x,t))|)}. \tag{6.31}$$

We follow these mesh parameters as described in [NT90] for of the NT and NTI schemes and in [PPR05] for the CWIRK2 and CWIRK3 schemes. We summarize the list below. For $\lambda \leq \lambda_0$,

$$\begin{aligned} \text{NT and NTI:} \quad & \lambda_0 = 0.32, \\ \text{CWIRK2:} \quad & \lambda_0 = \frac{1}{2}, \\ \text{CWIRK3:} \quad & \lambda_0 = \frac{3}{7}. \end{aligned} \tag{6.32}$$

We will use these parameters for our numerical experiments in Chapter 7.

Chapter 7

Numerical Results

In order to show the behavior and accuracy of our piecewise linear and piecewise quadratic reconstructions compared with the NT scheme that are presented in Chapter 5 and Chapter 6, some test problems with known analytical solution are solved in this chapter. In the scalar case, a linear advection equation with smooth initial condition will be tested to show the order of accuracy of the schemes. Burgers' equation and Buckley-Leverett's equation will be studied to see the behavior of the methods. We also test the total variation of the numerical experiments. We then study the applications of the schemes to systems of conservation laws. The classical tests from gas dynamics concerning Euler's equations for an ideal gas will be discussed.

We denote the methods as follows:

- **NT**: The Nessyahu and Tadmor scheme [NT90].
- **NTI**: The NT occupied with linear IMLS method, see (6.16)- (6.18).
- **CWIRK2**: the Central WENO-IMLS method based on the piecewise linear polynomial for discretization in space and the Runge-Kutta for time integration.
- **CWIRK3**: the Central WENO-IMLS method based on the piecewise quadratic polynomial for discretization in space and the Runge-Kutta for time integration, see (6.24)- (6.27).

7.1 Linear advection equation

7.1.1 Smooth initial data

To test the accuracy, we start with a linear advection equation with smooth initial data $u(x, t = 0) = \sin^4(\pi x)$. The computation is stopped at time $T = 1$. This test is used to test the convergence rate of the schemes.

TABLE 7.1
NT scheme

N	L_1 error	L_1 order	L_∞ error	L_∞ order
20	3.79E-002	-	1.27E-001	-
40	1.31E-002	1.53	5.73E-002	1.14
80	4.19E-003	1.64	2.53E-002	1.18
160	1.18E-003	1.83	1.06E-002	1.26
320	3.19E-004	1.89	4.32E-003	1.29

TABLE 7.2
NTI scheme

N	L_1 error	L_1 order	L_∞ error	L_∞ order
20	3.89E-002	-	1.29E-001	-
40	1.37E-002	1.51	5.67E-002	1.18
80	4.74E-003	1.53	2.45E-002	1.21
160	1.53E-003	1.63	9.94E-003	1.30
320	5.00E-004	1.61	3.94E-003	1.33

TABLE 7.3
CWIRK2 scheme

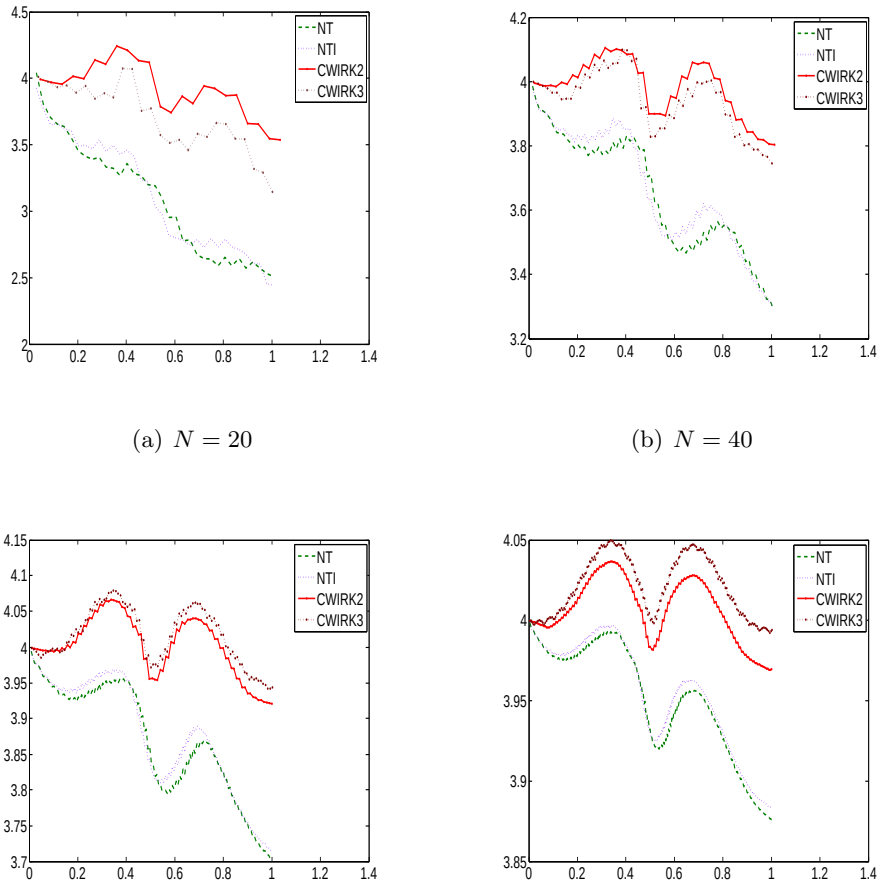
N	L_1 error	L_1 order	L_∞ error	L_∞ order
20	6.84E-002	-	1.06E-001	-
40	2.30E-002	1.57	4.49E-002	1.25
80	6.50E-003	1.82	1.97E-002	1.19
160	1.50E-003	2.11	7.70E-003	1.35
320	2.31E-004	2.70	1.50E-003	2.36

TABLE 7.4
CWIRK3 scheme

N	L_1 error	L_1 order	L_∞ error	L_∞ order
20	1.27E-001	-	2.03E-001	-
40	3.63E-002	1.81	7.32E-002	1.47
80	5.40E-003	2.75	1.54E-002	2.25
160	4.76E-004	3.50	1.60E-003	3.27
320	1.98E-004	2.33	3.87E-004	2.26

The computational parameters used in this test are $\epsilon = 10^{-4}$. The mesh ratio was chosen as $\lambda = 0.9\lambda_0$ where λ_0 is selected in order to satisfy stability conditions and it depend on the scheme. The choice of this parameter is presented in Chapter 6, subsection 6.3.3.

Table 7.1- 7.4 show the numerical errors and accuracy for the discrete L_1 norm and L_∞ norm where N is the number of grid points. The behavior of the errors and the order of accuracy is comparable for all four methods. It can be seen that the NT, NTI, and CWIRK2 schemes are second order accurate. The CWIRK3 schemes are third order accurate but they loose the accuracy on the fine grids. Concerning the NT and NTI schemes, we can see that the accuracy of the NT scheme performs better than the NTI schemes because the difference operators of the first derivatives



(a) $N = 20$ (b) $N = 40$

(c) $N = 80$ (d) $N = 160$

FIGURE 7.1 *Total Variation of approximations of linear advection equation with smooth initial data*

in the NTI scheme do not produce an optimal second order accuracy, see Chapter 3, subsection 3.1.1.

We also test the convergence of the methods by plotting the numerical values of *Total Varialtion* (TV) presented in Chapter 4, subsection 4.2.2 of the discrete solution. The discrete version of TV is

$$TV(u) := \sum_{j=1}^{N} |u_{j+1} - u_j|. \tag{7.1}$$

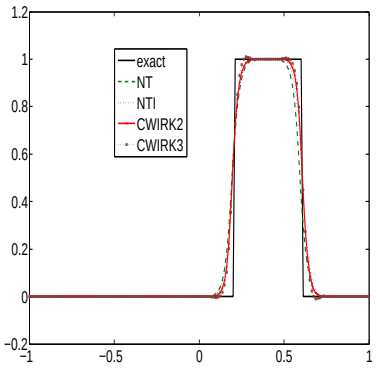
In Figure 7.1, we show the TV of four numerical schemes with the different numbers of grid points. The sizes of the TV of all four schemes are decreased as the grid refined. Nessyahu and Tadmor [NT90] pointed out that the scheme in (6.17) under the CFL condition, $\lambda = 0.32$, is TVD.

7.1.2 The square wave

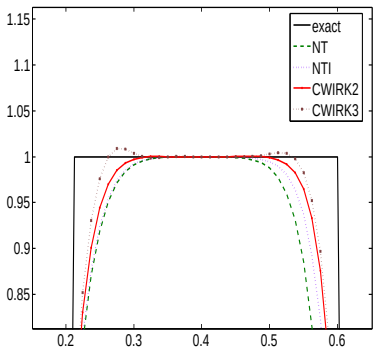
We consider a linear advection equation with the initial data given by

$$u(x, t = 0) = \begin{cases} 1 & -0.8 \leq x \leq 0.4, \\ 0 & \text{otherwise.} \end{cases}$$

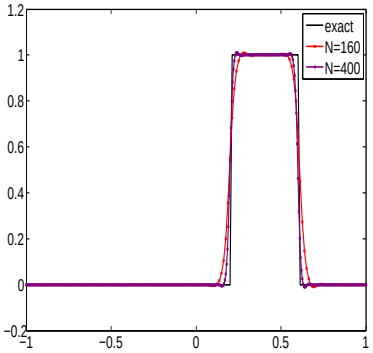
The numerical result at time $T = 1$ of this problem is shown in Figure 7.2. In Figure 7.2(b) we observe that the results of the NTI, CWIRK2, CWIRK3 schemes are sharper than the result of the NT scheme but the CWIRK3 scheme generates oscillations at the top of the wave. Figure 7.2(c) shows the result obtained with the CWIRK3 for two grid sizes, $N = 160$ and $N = 400$. The solution has spurious oscillations even though the grid is refined. Figure 7.3 confirms that although the grid size is small, the TV of the CWIRK3 is increased for $N = 80$ and $N = 160$.



(a) four different schemes

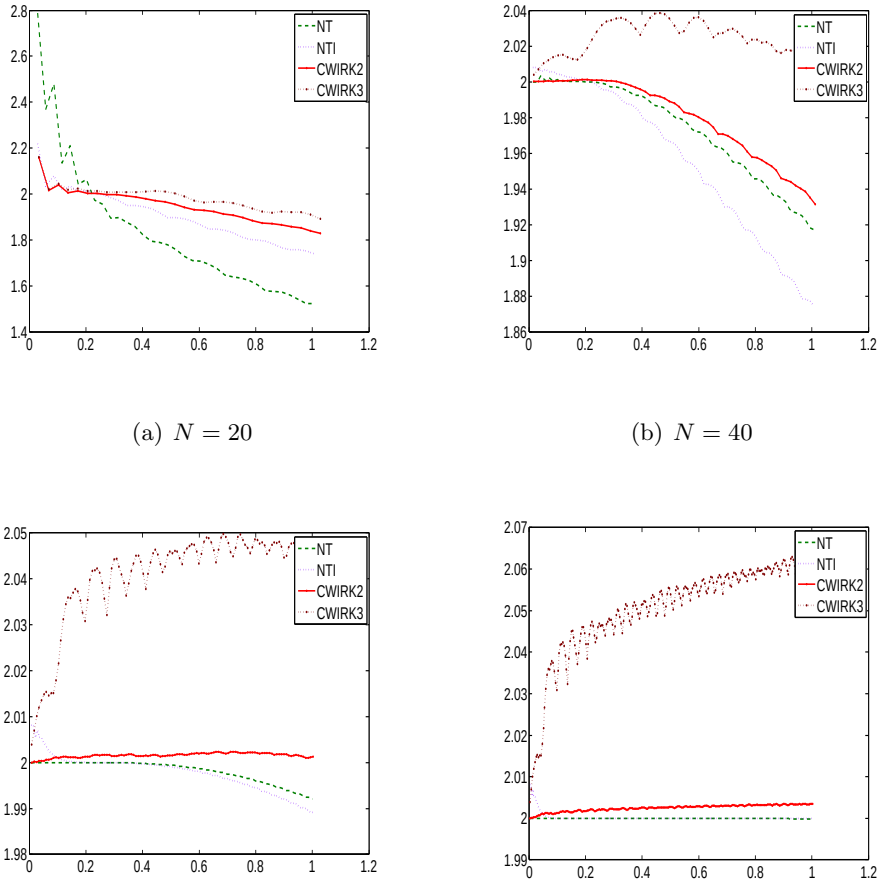


(b) Zoomed region



(c) CWIRK3

FIGURE 7.2 *Linear advection equation with non-smooth initial data*



(a) $N = 20$ (b) $N = 40$

(c) $N = 80$ (d) $N = 160$

FIGURE 7.3 *Total Variation of approximations of linear advection equation with the square wave data*

7.2 Burgers' Equation

Consider Burgers' equation

$$u_t + \left(\frac{1}{2}u^2\right)_x = 0$$

with initial condition

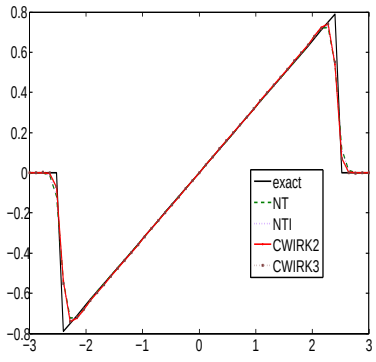
$$u(x, t = 0) = \begin{cases} \frac{x}{2} & -2 \leq x \leq 2, \\ 0 & \text{otherwise} \end{cases}$$

on the interval $[-3, 3]$. The exact solution is given by

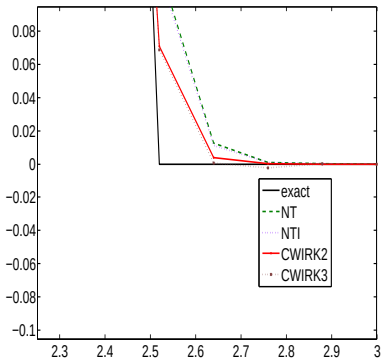
$$u(x,t) = \begin{cases} \frac{x}{t+2} & -\sqrt{2(t+2)} \leq x \leq \sqrt{2(t+2)}, \\ 0 & \text{otherwise.} \end{cases} \tag{7.2}$$

This solution has two shock propagating with the speeds $\pm\frac{1}{\sqrt{2(t+2)}}$. The right-going shock has the left state $u_l = \sqrt{\frac{2}{t+2}}$ and the right state $u_r = 0$ and similarly for the left-going shock.

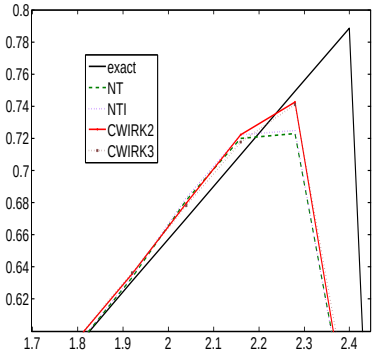
Figure 7.4(a)- 7.4(c) show the results of Burgers' equation at time $T = 1$. The numerical demonstration from the NT and NTI schemes shows that the NT and NTI have the same behavior. And the difference between numerical solutions obtained with the CWIRK2 and CWIRK3 is very small shown in Figure 7.4(b)- 7.4(c). As expected, the CWIRK3 scheme acts as the second order reconstruction in the presence of shocks. Figure 7.5 contains various plot of the TV approximations of four schemes for the different grids, $N = 20, N = 40, N = 80, N = 160$. It seems that the size of the TV of all four schemes are increased although the grid is refined.



(a) four different schemes

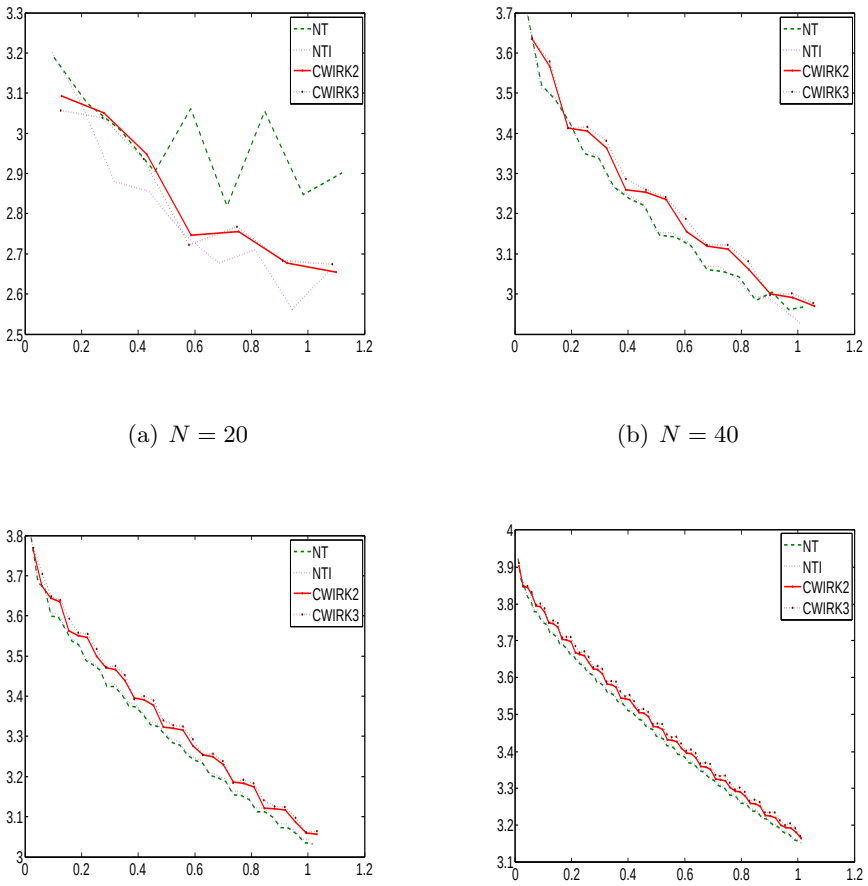


(b) Zoomed region



(c) Zoomed region

FIGURE 7.4 *Burgers' Equation at* $T = 1$



(a) $N = 20$ (b) $N = 40$

(c) $N = 80$ (d) $N = 160$

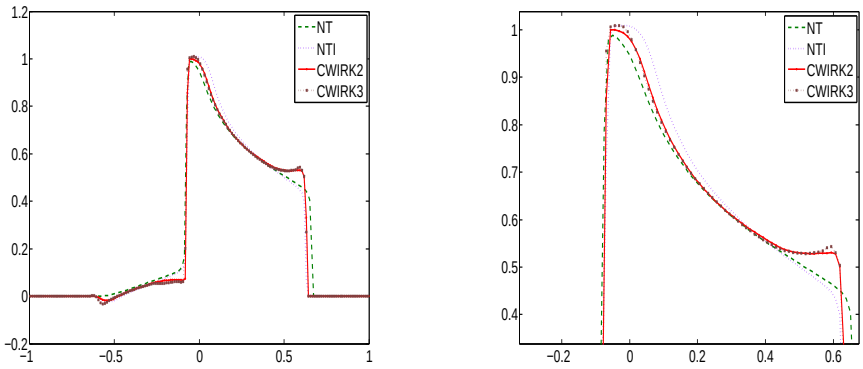
FIGURE 7.5 *Total Variation of approximations of Burgers' equation*

7.3 Buckley-Leverett's problem

$$u_t + f(u)_x = 0, \quad f(u) = \frac{4u^2}{4u^2 + (1-u)^2}, \quad -1 \le x \le 1$$

subject to the initial data

$$u(x, t=0) = \begin{cases} 1 & -\frac{1}{2} \le x \le 0, \\ 0 & \text{otherwise.} \end{cases}$$



(a) four different schemes (b) Zoomed region

FIGURE 7.6 *Buckley-Leverett's equation*

We have computed the solution up to time $T = 0.4$ and $N = 160$. Figure 7.6 shows the results obtained with four different schemes. As the plot show, the results from the CWIRK2 and CWIRK3 are very similar and the behaviour of the NT and NTI are slightly different.

7.4 Euler equations

The system of Euler equations of gas dynamics for a polytropic gas with $\gamma = 1.4$ is considered. The variables ρ, m, E and p, which will be shown below, are the density, momentum, total energy per unit volume, and the pressure, respectively. In this work, the componentwise extension will be adopted. With four numerical methods the smoothness indicators are computed using the formulas as for the scalar case for each component. Here, we apply a general strategy to compute the time step,

$$\Delta t = \frac{0.5\lambda_0 h}{\max_j(c_j + |u_j|)}, \tag{7.3}$$

where c_j and u_j are the local sound speed and velocity respectively.

7.4.1 Sod's initial data

The first test problem has the initial values:

$$(\rho, m, E) = \begin{cases} (1, 0, 0.25) & x \leq 0.5, \\ (0.125, 0, 0.25) & x > 0.5. \end{cases}$$

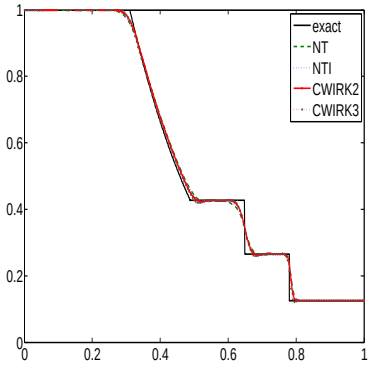
The computation domain is $[0, 1]$ and time integration is up to $T = 0.16$, and $N = 200$. The exact solution of Sod's initial data is described by B. Fryxell [Fry]. The solution to this problem contains one rarefaction wave traveling to the left, one shock wave traveling to the right, and a contact discontinuity in between.
Figure 7.7 shows results with four different schemes. It can be seen that the behavior of the NT and NTI is similar. The difference between the numerical solutions obtained with the CWIRK2 and CWIRK3 is very small. Because the reconstruction of the CWIRK3 is composed of a quadratic polynomial and two one-sided linear polynomials, if the non-linear weights are not optimal, the two linear functions are not balanced by the central quadratic function. Therefore, the CWIRK3 reconstruction is switched to one of the one-side linear reconstruction.

7.4.2 Lax's initial data

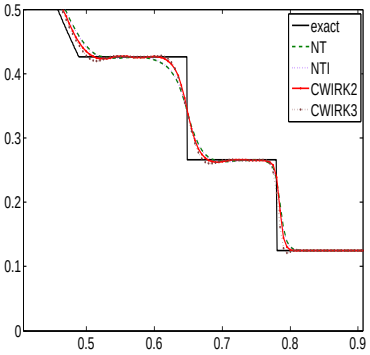
This test problem is the one other example of a Riemann problem for gas dynamics. The initial data is

$$(\rho, m, E) = \begin{cases} (1, -19.59745, 2692.03) & x \leq 0.8, \\ (1, -19.59745, 192.05) & x > 0.8. \end{cases}$$

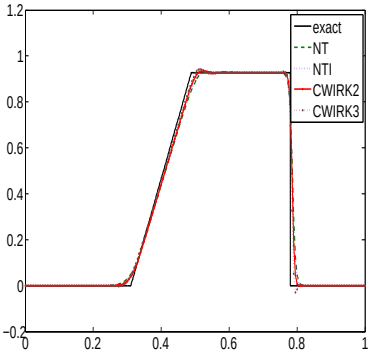
The solution of this problem contains two shock waves and a contact discontinuity in between. The computation domain is $[0, 1]$ and time integration is up to $T = 0.012$, and $N = 200$. The exact solution of this problem is obtained by Sutherland [Sut]. The results are illustrated in Figure 7.8. Similar to the Sod's problem, the results from the NT and NTI are not different. In the velocity profile shown in Figure 7.8(c), all four schemes produce very similar results.



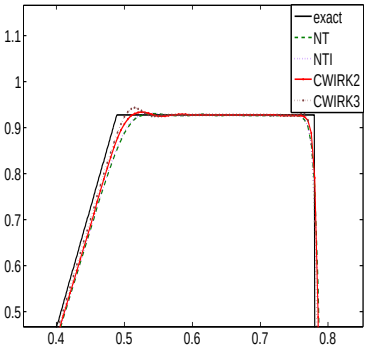
(a) density



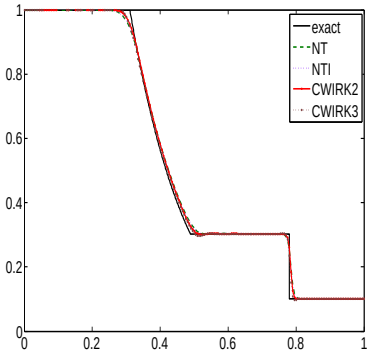
(b) zoomed density



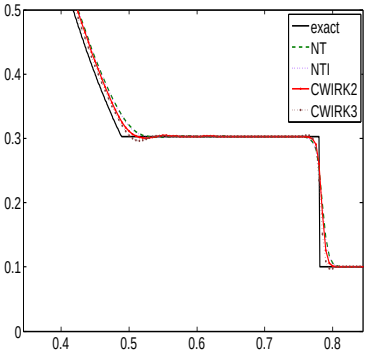
(c) velocity



(d) zoomed velocity

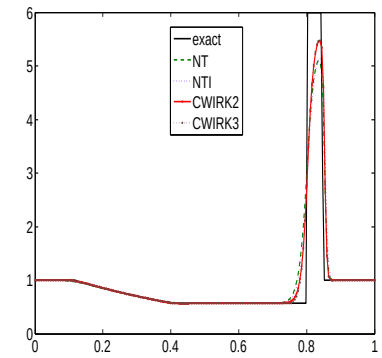


(e) pressure

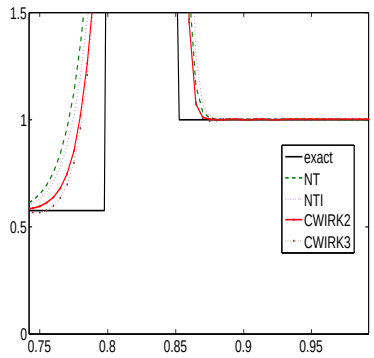


(f) zoomed pressure

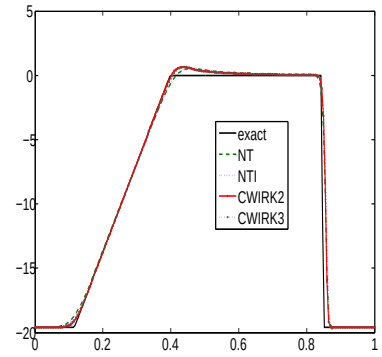
FIGURE 7.7 *Sod problem*



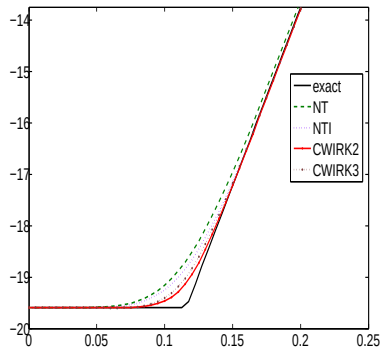
(a) density



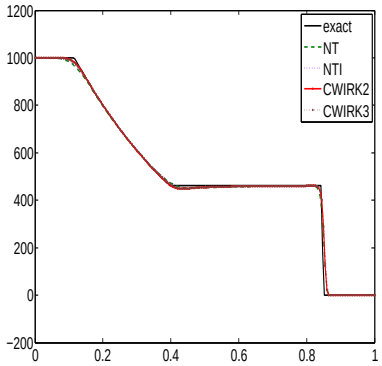
(b) zoomed density



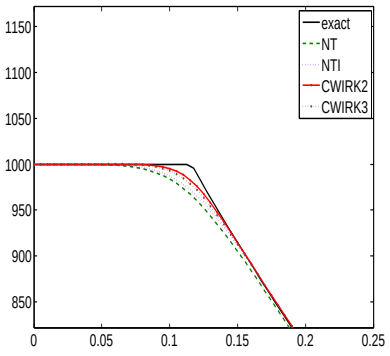
(c) velocity



(d) zoomed velocity



(e) pressure



(f) zoomed pressure

FIGURE 7.8 *Lax problem*

Chapter 8

Summary and Outlook

8.1 Summary

In this thesis, the numerical methods based on the meshfree IMLS method and the central WENO techniques have been developed for solving hyperbolic conservation laws. In order to achieve this we have carried on as follows.

An overview of meshfree methods has been presented. In this study, the MLS approximation and its interpolating version are only discussed for constructing the meshfree shape functions. The other approach -RKPM- is out of the scope of this study. In order to apply the upwind strategies in our new schemes to hyperbolic problems, the non-symmetric window functions have also been introduced for computing the IMLS interpolants. The explicit expression of the IMLS methods based on symmetric and non-symmetric window functions has been descreibed. These formulae are very useful to analyse the L_2 error of our new reconstructions based on the IMLS methods. Von Neumann stability analysis of the IMLS methods for a linear advection equation has been presented.

High order schemes have been constructed using the derivatives of the IMLS methods in a Taylor expansion. Both the piecewise linear and quadratic reconstructions have been proposed. The common obstacle of high order schemes is the presence of the unwanted oscillations. To eliminate this, the idea of the WENO schemes has been adopted. We have taken a convex combination of two linear functions for the second order schemes and designed the proper weights to prevent the oscillations. The quadratic reconstruction has been formulated using a combination of two linear functions and a parabolic function. If the solution is on the smooth region, the method is of the third order. In turn, the method is of the second order near discontinuities or the presence of large gradients. We have introduced the central Runge-Kutta scheme for time integration to solve the problems. For the purpose of comparison, the NT scheme has been discussed. We have also applied the IMLS methods in the NT scheme. In this framework, the minmod limiter has been used

to prevent the oscillations.

Numerical results from the four schemes have been presented. As expected, all four schemes is of the second order if the solution is near discontinuties or extrema. But in smooth region the quadratic reconstruction is of the third order whereas the other schemes are of the second order.

8.2 Outlook

The other approach of the third order central schemes to achieve more accurate and stable solutions is to contruct the schemes based on the Godunov method as in [LT98]. The main idea is using a piecewise quadratic polynomial reconstruction which enjoys non-oscillatory property and employing a proper quadrature rule to approximate the numerical fluxes. Because of this, the method is of the third order accuracy on the entire domain.

To improve more accuracy, a piecewise cubic polynomial in which its successive derivatives are computed using the IMLS methods can be reconstructed. To fulfill non-oscillatory property, the minmod limiter is introduced in the method. The details of this concept are given in [PGDB08] for the mesh-based method.

Bibliography

[AD01] M. G. Armentano and R. G. Duän. Error estimates for moving least square approximations. *Appl. Num. Math.*, 37:397–416, 2001.

[AL01] N. R. Aluru and G. Li. Finite cloud method: A true meshless technique based on a fixed reproducing kernel approximation. *Int. J. Numer. Meth. Engng*, 50:2373–2410, 2001.

[Alu00] N. R. Aluru. A point collocation method based on reproducing kernel approximations. *Int. J. Numer. Meth. Engng*, 47:1083–1121, 2000.

[ATP84] D. A. Andeson, J. C. Tannehill, and R. H. Pletcher. *Computational Fluid Mechanics and Heat Transfer.* Hemisphere Publishing Corporation, 1984.

[AZ98] S. N. Atluri and T. Zhu. A new meshless local petrov-galerkin (MLPG) approach in compuational mechanics. *Comput. Mech.*, 22:117–127, 1998.

[BKFK96] T. Belytschko, Y. Krongauz, M. Fleming, and P. Krysl. Meshless methods: An overview and recent developments. *Com. Meth. App. Mech. Eng.*, pages 3–47, 1996.

[BLG96] T. Belytschko, Y. Y. Lu, and L. Gu. Dynamic frature using element free galerkin methods. *Int. J. Numer. Meth. Engng*, 39:923–938, 1996.

[BOK95] T. Belytschko, D. Organ, and Y. Krongauz. A coupled finite elemet free galerkin. *Comput. Mech.*, 17:186–195, 1995.

[BPR99] F. Bianco, G. Puppo, and G. Russo. High order central schemes for hyperblic systems of conservation laws. *SIAM J. Sci. Comput*, 21:294–322, 1999.

[BT96] T. Belytschko and M. Tabbara. Dynamic fracture using element-free galerkin methods. *Int. J. Numer. Meth. Engng*, 39:923–938, 1996.

[BUG01] J. J. Benito, F. Urena, and L. Gavete. Influence of several factors in the generalized finite difference method. *Appl. Math. Modelling*, 25:1039–1053, 2001.

[BUG07] J. J. Benito, F. Urena, and L. Gavete. Solving parabolic and hyperbolic equations by the generalized finite difference method. *J. Comput. Appl. Math.*, 209:208–233, 2007.

[CFC08] L. Cueto-Felgueroso and I. Colominas. High-order finite volume methods and multiresolution reproducing kernels. *Arch. Comput. Methods Eng.*, 15:185–228, 2008.

[CFCF+06] L. Cueto-Felgueroso, I. Colominas, J. Fe, F. Navarrina, and M. Casteleiro. High order finite volume schemes on unstructured grids using moving least squares reconstruction. Application to shallow water dynamics. *Int. J. Numer. Meth. Engng.*, 65:295–331, 2006.

[CFCN+07] L. Cueto-Felgueroso, I. Colominas, X. Nogueira, F. Navarrina, and M. Casteleiro. High order finite volume schemes on unstructured grids using moving least squares reconstruction. application to shallow water dynamics. *Comput. Methods Appl. Mech. Engng.*, 196:4712–4736, 2007.

[CM97] E. Chow and J. J. Monaghan. Ultrarelativistic SPH. *J. Comput. Phys.*, 134:296–305, 1997.

[CM99] P. W. Cleary and J. J. Monaghan. Conducing modelling using smoothed particle hydrodynamics. *J. Comput. Phys.*, 148:227–264, 1999.

[CSC] CSCAMM. Central station: A collection of references on high-resolution non-oscillatory central schemes. Website. Available online at`http://www.cscamm.umd.edu/centpack/publications`; visited on 30.10.2009.

[Fas07] G. E. Fasshauer. *Meshfree Approximation Methods with Matlab.* World Scienctific Publishing Co. Pte. Ltd, 2007.

[Fri98] O. Friedrich. Weighted essentially non-oscillatory schemes for the interpolation of mean values on unstructured grids. *J. Comput. Phys.*, 144:194–212, 1998.

[Fri99] O. Friedrich. *Gewichtete Wesentlich Nicht-Oszillierende Verfahren auf Unstrukturierten Gittern.* PhD dissertation, Universität Hamburg, Fakutät für Mathematik, July 1999.

[Fri03] T.P. Fries. Classification and overview of meshfeee method. Technical report, Institut für Wissenschaftliches Rechnen , TU Braunschweig, 2003.

[Fry] B. Fryxell. Solves the exact riemann problem. `http://cococubed.asu.edu/codes/riemann/exact_riemann.f`. Online 10.03.2010.

[FS01] J. Fürst and Th. Sonar. On meshless collocation approximations of conservation laws: Preliminary inverstigations on positive schemes and dissipation methods. *ZAMM*, 81:403–415, 2001.

[Gho10] M. Ghorbani. Diffuse element kansa method. *Appl. Mathe. Sci.*, 4(12):583–594, 2010.

[GL05] Y. T. Gu and G. R. Liu. A meshfree weak-strong (MWS) form method for time dependent problems. *Comput. Mech.*, 35:134–145, 2005.

[GM77] R. A. Gingold and J. J. Monaghan. Smoothed particle hydrodynamics: Theory and application to non-spherical stars. *MNRAS*, 181:375–389, 1977.

[GR96] E. Godlewski and P. A. Raviart. *Numerical Approximations of Hyperbolic Systems of Conservation Laws.* Springer, New York, 1996.

[Heg96] D. Hegen. Element-free galerkin methods in combination with finite element approaches. *Com. Meth. App. Mech. Eng.*, 135:143–166, 1996.

[HEOC87] A. Harten, B. Engquist, S. Osher, and S. Chakravarthy. Uniformly high order essentially non-oscillatory scheme, III. *J. Comput. phy.*, 71:231–303, 1987.

[Hir90] C. Hirsch. *Numerical Computation of Internal and External Flows: Vol.2.* John Wiley and Sons, 1990.

[HLRA06] Z. D. Han, H. T. Liu, A. M. Rajendran, and S. N. Atluri. The applications of meshless local petrov-galerkin (MLPG) approaches in high-speed impact, penetration and perforation problems. *CMES*, 14(2):119–128, 2006.

[IOCP03] S. R. Idelsohn, E. Onate, N. Calvo, and F. D. Pin. The meshless finite element method. *Int. J. Numer. Meth. Engng*, 58:893–912, 2003.

[JS96] G. S. Jiang and C. W. Shu. Efficient implementation of weighted ENO schemes. *J. Comput. phy.*, 126(0130):202–228, January 1996.

[Kan90a] E. J. Kansa. Multiquadratics- a scattered data approximation scheme with applications to computational fluid dynamics: I. surface approximations and partial derivative estimates. *Comput. Math. Appl.*, 19:127–145, 1990.

[Kan90b] E. J. Kansa. Multiquadratics- a scattered data approximation scheme with applications to computational fluid dynamics: II. solutions to parabolic, hyperbolic, and elliptic partial differential equations. *Comput. Math. Appl.*, 19:147–161, 1990.

[KB95] P. Krysl and T. Belytschko. Analysis of thin plates by the element free galerkin methods. *Comput. Mech.*, 17:26–35, 1995.

[KB96] Y. Krongauz and T. Belytschko. Enforcement of essential boundary conditions in meshless approximations using finite elements. *Com. Meth. App. Mech. Eng.*, 131:133–145, 1996.

[KT00] A. Kurganov and E. Tadmor. New high-resolution central schemes for nonlinear conservation laws and convection-diffusion equations. *J. Comput. Phy.*, 160:241–282, 2000.

[Kun01] M. Kunle. *Entwicklung und Untersuchung von Moving Least Square Verfahren zur numerischen Simulation hydrodynamischer Gleichungen.* PhD dissertation, Eberhard-Karls-Universität zu Tübingen, Fakutät für Physik, November 2001.

[LA01] H. Lin and S. N. Atluri. The meshless local petrov-galerkin (MLPG) methods for solving incompressible navier-stokes equations. *CMES*, 2(2):117–142, 2001.

[Lan98] C. B. Laney. *Computational Gasdynamics.* Cambridge University Press, 1998.

[LBG92] Y. Y. Lu, T. Belytschko, and L. Gu. A new implementation of the element free galerkin method. *Com. Meth. App. Mech. Eng.*, 113:397–414, 1992.

[LC02] Y. Luo and U. H. Combe. A generalized finite-difference method based on minimizing global residual. *Com. Meth. App. Mech. Eng.*, 191:1421–1438, 2002.

[LCJ+96] W. K. Liu, Y. Chen, S. Jun, J. S. Chen, T. Belytschko, C. Pan, Uras R. A, and C. T. Chang. Overview and applications of the reproducing kernel particle methods. *Arch. Comput. Method. E.*, 3:3–80, 1996.

[LDT96] T. J. Liszka, C. A. M. Duarte, and W. W. Tworzydlo. Hp-meshless cloud method. *Com. Meth. App. Mech. Eng.*, 139:263–288, 1996.

[LeV99] R. J. LeVeque. *Numerical Methods for Conservation Laws.* Birkhaeuser Verlag, 1999.

[LeV02] R. J. LeVeque. *Finite Volume Methods for Hyperbolic Problems.* Cambridge University Press, 2002.

[LG04] G. R. Liu and Y. T. Gu. A meshfree method: Meshfree weak-strong (MWS) form method for 2-d solids. *Comput. Mech.*, 35:134–145, 2004.

[LHC+97] W. K. Liu, W. Hao, Y. Chen, S. Jun, and J. Gosz. Multiresolution reproducing kernel particle methods. *Comput. Mech.*, 20:295–309, 1997.

[LJZ95] W. K. Liu, S. Jun, and Y. F. Zhang. Reproducing kernel particle methods. *Int. J. Numer. Methods Fluids*, 20:1081–1106, 1995.

[LL99] S. Li and W. K. Liu. Reproducing kernel hierarchical partition of unity, part i-formulation and theory. *Int. J. Numer. Methods*, 45:251–288, 1999.

[LL02] S. Li and W. K. Liu. Meshfree and particle methods and their applications. *Appl. Mech. Rev.*, 55(1):1–34, 2002.

[LLB97] W. K. Liu, S. Li, and T. Belytschko. Moving least square reproducing kernel method part I: Methodology and convergence. *Com. Meth. App. Mech. Eng.*, 143:113–154, 1997.

[LLL97] M. B. Liu, G. R. Liu, and K. Y. Lam. Computer simulation of high explosive using smoothed particle hydrodynamics methodology. *Comput. Fluids*, 32(3):305–322, 1997.

[LO80] T. Liszka and J. Orkisz. The finite-difference method at arbitraty irregular grids and its application on applied mechanics. *Comput. Struct.*, 11:83–95, 1980.

[LOC94] X. D. Liu, S. Osher, and T. Chan. Weighted essentially non-oscillatory schemes. *J. Comput. Phy.*, 115:200–212, April 1994.

[LPR99a] D. Levy, G. Puppo, and G. Russo. Central weno schemes for hyperbolic systems of conservation laws. *ESAIM: M2AN*, 33:547–571, 1999.

[LPR99b] D. Levy, G. Puppo, and G. Russo. Compact central weno schemes for multidimensional conservation laws. *SIAM J. Sci. Comput.*, 22:656–672, 1999.

[LPR00] D. Levy, G. Puppo, and G. Russo. On the behavior of the total variation in cweno methods for conservation laws. *Appl. Numer. Math.*, 33:407–414, 2000.

[LRR99] S. F. Liotta, V. Romano, and G. Russo. *Central Schemes and Systems of Balnce Laws*, volume 130 of *Hyperbolic Problems: Theory, Numerics, Applications.* Birkhuser Verlag Basel/Switzerland, 1999.

[LS81] P. Lancaster and K. Salkauskas. Surface generated by moving least squares method. *Math. Comp.*, 37:142–158, 1981.

[LS86] P. Lancaster and K. Salsauskas. *Curve and Surface Fitting: An Introduction.* Academic Press, 1986.

[LT98] X. D. Liu and E. Tadmor. Third order nonoscillatory central scheme for hyperbolic conservation laws. *Numer. Math.*, 70:397–425, 1998.

[Luc93] L. B. Lucy. Numerical approach to testing of fission hypothesis. *Astron. J.*, 82:1475–1478, 1993.

[MCM77] Y. Marechal, J. L. Coulomb, and G. Meunier. Use of the diffuse element method for electromagnetic field computation. *IEEE Trans. Magn.*, 29(2):1013–1024, 1977.

[MFZ97] J. P. Morris, P. J. Fox, and Y. J. Zhu. Modelling low Reynolds number incompressible flows using SPH. *J. Comput. Phys.*, 136:214–226, 1997.

[MM94] K. W. Morton and D. F. Mayers. *Numerical Solution of Partial Differential Equations.* Cambridge University Press, 1994.

[Mon94] J. J. Monaghan. Simulation free surface flows with SPH. *J. Comput. Phys.*, 110:399–406, 1994.

[Mon05] J. J. Monaghan. Smoothed particle hydrodynamics. *Rep. Prog. Phys.*, 68:1703–1759, 2005.

[NCFC+08] X. Nogueira, L. Cueto-Felgueroso, I. Colominas, S. Khelladi, F. Navarrina, and M. Casteleiro. A finite volume method using moving least squares approximations for the resolution of acoustic problems on unstructured grids. ACOMEN, University of Liege, Belgium, May 2008.

[Net08] H. Netuzhylov. *A Space-Time Meshfree Collocation Method for Coupled Problems on Irregularly-Shaped Domains.* PhD dissertation, Technische Universität Braunschweig, Computational sciences in engineering, 2008.

[Nid05] M. Von Nida. *Meshfree Methods for the Dynamics of Solids.* PhD dissertation, Technische Universität Kaiserslautern, Fakutät für Mathematik, April 2005.

[NSY07] H. Netuzhylov, Th. Sonar, and W. Yomsatieankul. Finite difference operators from moving least squares interpolation. *ESAIM: M2AN*, 41:959–974, 2007.

[NT90] H. Nessyahu and E. Tadmor. Non-oscillatory central differencing for hyperbolic conservation laws. *J. Comput. Phy.*, 87:408–463, 1990.

[NTV92] B. Nayroles, G. Touzot, and P. Villon. Generalizing the finite element method: Diffuse approximation and diffuse elements. *Comp. Mech.*, 10:307–318, 1992.

[OIZT96] E. Onate, S. Idelsohn, O. C. Zoenkieqicz, and R. L. Taylor. A finite point method in computational mechanics: Applications to convective transport and fluid flow. *Int. J. Numer. Methods*, 39:3839–3866, 1996.

[PFTV92] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling. *Numerical Recipes in C: The Art of Scientific Computing.* Cambridge University Press, 1992.

[PGDB08] A. I. Peer, A. Gopaul, M.Z. Dauhoo, and M. Bhuruth. A new fourth-order non-oscillatory central scheme for hyperbolic conservation laws. *Appl. Numer. Math.*, 58:674–688, 2008.

[PPR05] L. Pareschi, G. Puppo, and G. Russo. Central runge-kutta schemes for conservation laws. *SIAM J. Sci. Comput.*, 2005.

[PR06] G. Puppo and G. Russo. Staggered finite difference schemes for conservation laws. *SIAM J. Sci. Comput.*, 27:403–418, 2006.

[Son05] Th. Sonar. Difference operators from interpolating moving least squares and their deviation from optimality. *ESAIM: M2AN*, 39:883–908, 2005.

[SP96] H. Sadat and C. Prax. Application of the diffuse approximation for solving fluid flow and heat transfer problems. *Int. J. Heat Mass Transfer*, 39(1):214–218, 1996.

[Sut] Ralph Sutherland. 1d riemann tests. Available online at `http://www.mso.anu.edu.au/fyris/lw1driemann.html`; visited on 10.03.2010.

[Tho99] J. W. Thomas. *Numerical Partial Differential Equations: Conservation Laws and Elliptic Equations.* Springer-Verlag, 1999.

[Tra09] J. A. Trangenstein. *Numerical Solution of Hyperbolic Partial Differential Equations.* Cambridge, 2009.

[WL02] J. G. Wang and G. R. Liu. A point interpolation meshless method based on radial basis functions. *Int. J. Numer. Methods*, 54:1623–1648, 2002.

[ZA98] T. Zhu and S. N. Atluri. A modified collocation method and a penalty formulation for enforcing the essential boundary conditions in the element free galerkin method. *Comput. Mech.*, 21:211–222, 1998.

[ZB04] G. M. Zhang and R. C. Batra. Modified smoothed particle hydrodynamics method and its application to transient problems. *Comput. Mech.*, 34:137–146, 2004.

[ZYZ06] X. Zhang, Z. Yao, and Z. Zhang. Application of MLPG in large deformation analysis. *Acta Mech. Sin.*, 22:331–340, 2006.